Sweet Karoline's Cakes

ELABORATION PHASE

Logan Amlung | Chris Boaz | Kevin Bush | Brad Eadens
Ben Lascurain | Hannah Parker

TABLE OF CONTENTS

System Requirements

The system Requirements provides a hierarchy of both nonfunctional and functional requirements that the system will need in order to perform its designated functions. The requirements are separated into categories based on the functions primary user and are again broken down into Nonfunctional and Functional categories.

NONFUNCTIONAL REQUIREMENTS

Nonfunctional requirements refer to behavioral properties that the system must have and affect decisions made during the design of the system. Requirements in category 1 correspond to business hardware and software ability to operate in tandem. Requirements in category 2 correspond to the system's ability to store information on quotes, payments, orders, and customers. Requirements in category 3 correspond to security features. The requirements in category 4 correspond to functions involving any potential cultural and political requirements that come from running a small business.

FUNCTIONAL REQUIREMENTS

Functional requirements relate directly to a process a system has to perform or information that it needs to contain. Requirements in category 1 correspond to business customer's ability to submit a quote request form with required and optional information. Requirements in category 2 correspond to managing the schedule to ensure appointments are made and managed properly. Requirements in category 3 correspond to producing a manager validated schedule for the week. The requirements in category 4 correspond to creating customer profiles for website login and capturing information. The requirements in category 5 correspond to accounting practices.

SYSTEM REQUIREMENTS

Nonfunctional Requirements
1. Operational Requirements
1.1 The system will be backed up every day at Close of Business
1.2 The system will be able to be recovered from backup.
2. Performance Requirements
2.1 The system will process Quote Request
2.2 The system will process customer payments
2.3 The system will store order request
2.4 The system will store customer information in a database
3. Security Requirements
3.1 Only management will have the capability to set/alter availability
3.2 Only management will have the capability to alter calendar events/schedule

Functional Requirements
1. Quote Request Form
1.1 Customer will submit for a Quote
1.2 Customer will upload applicable .jpeg file
2. Manage Schedule
2.1 Customer will select an appointment time/date
2.2 Customer can change appointment time/date
2.3 Customer can cancel appointment
3. Produce Schedule
3.1 Management will check weekly schedule
3.2 Management will confirm weekly schedule
3.3 Management will print out weekly schedule
4. Customer Profile
4.1 Customer will create a login ID
4.2 Customer will create a login password
4.3 Customer will enter payment information
4.4 Customer will enter contact information
4.5 Customer will enter address information
4.6 Customer will update customer information
4.7 Customer can recover forgotten password
5. Revenue / Expense Tracking Procedures
5.1 System will produce reports for a given period of time
5.2 System will print out reports
5.3 Management will submit cost
6. Create Email Campaign
6.1 System will generate template
6.2 System will populate customer email list
7. Product Interaction
7.1 Customer will view product info
7.2 Customer will view previous item
7.3 Customer will navigate to Social Media
8. Management Inventory
8.1 System will track ingredients
8.2 System will generate ingredients
8.3 Management will update ingredients

Trace Matrix

The following graphs are the use case matrix for functional and non-functional system requirements. The matrix shows which use cases are used to handle the corresponding system requirements. The use cases are listed vertically along the side and the system requirements are listed horizontally along the top.

Non-Functional Requirements Matrix:

	1.1	1.2	2.1	2.2	2.3	2.4	3.1	3.2
View previous items								
View product info								
Jump to social media								
create quote request								
Create order					X			
Submit payment				X				
Schedule completion date								
Change completion date								
Cancel completion date								
Receive quote request			X					
Check weekly Schedule								
Confirm weekly schedule							X	X
Print weekly schedule								
Create email campaign								
Generate marketing report								
Print marketing reports								
Create customer Account						X		
Forgot login info								
Generate backup	X							
Recover from backup		X						
Edit Customer Account								
Generate ingredient list								
Track inventory								
Update inventory								
Submit costs								
Generate financial reports								
Print financial reports								

Functional Requirements Matrix:

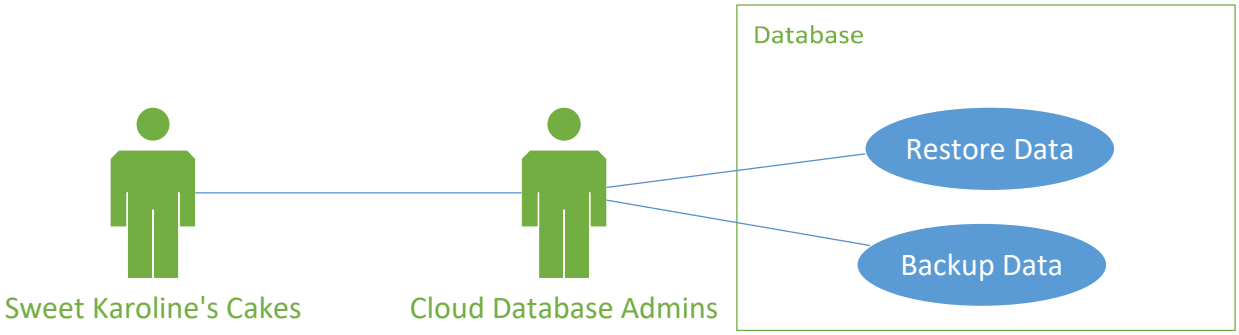
	1.1	1.2	2.1	2.2	2.3	3.1	3.2	3.3	4.1	4.2	4.3	4.4	4.5	4.6	4.7	5.1	5.2	5.3	6.1	6.2	7.1	7.2	7.3	8.1	8.2	8.3
View previous items																						X				
View product info																					X					
Jump to social media																							X			
Create quote request	X	X										X	X													
Create order												X	X													
Submit payment											X															
Schedule completion date			X																							
Change completion date				X																						
Cancel completion date					X																					
Receive quote request																										
Check weekly Schedule						X																				
Confirm weekly schedule							X																			
Print weekly schedule								X																		
Create email campaign																			X	X						
Generate marketing report																X										
Print marketing reports																	X									
Create customer Account									X	X		X	X													
Forgot login info															X											
Generate backup																										
Recover from backup																										
Edit Customer Account														X												
Generate ingredient list																									X	
Track inventory																								X		
Update inventory																										X
Submit costs																		X								
Generate financial reports																X										
Print financial reports																	X									

Use Case Diagram

A Use Case Diagram is composed of two main parts; the actor and the use case name. The actor can be any entity (often a person) that is associated with a use case. A use case name is the title of the use case that it represents. The use case itself describes the scenario of use, the preconditions, postconditions, special requirements, main flow, and other aspects. This diagram keeps it simple, so readers can see the relation between a use case and the actor without being bogged down by all the details.

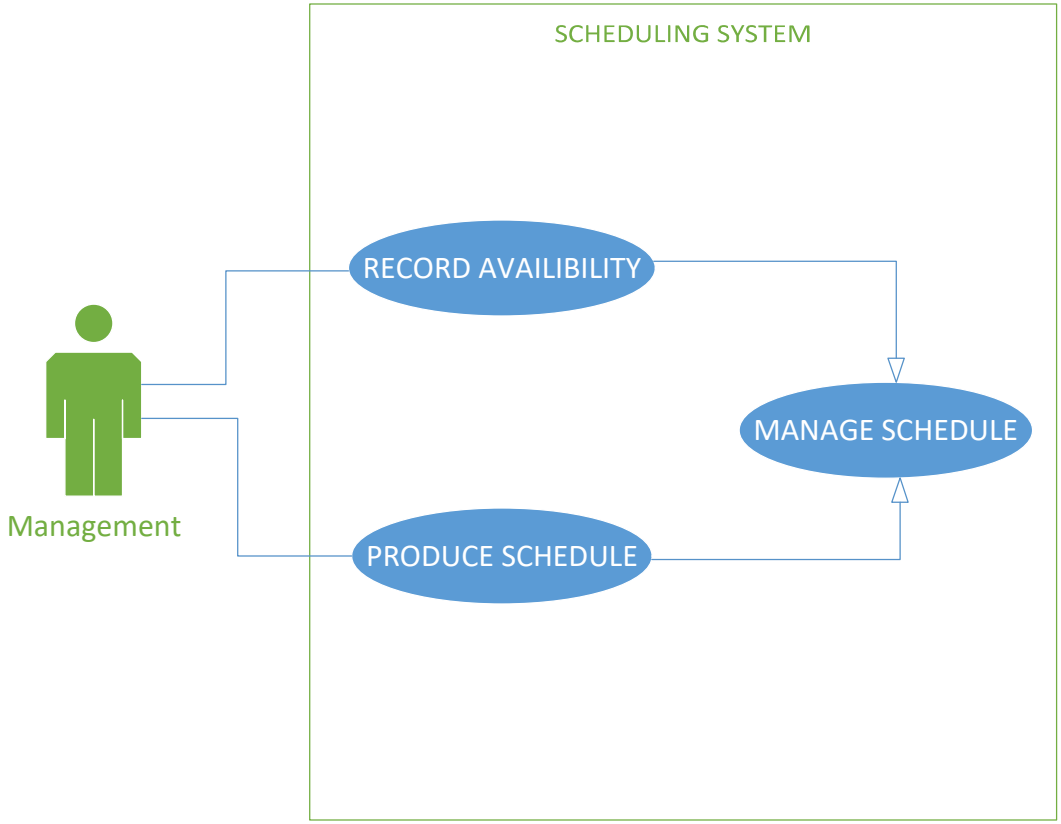
Database

The only actor for interacting directly with the database is the cloud database administrator. The DBA can set up automatic data backups and can restore data when requested by Sweet Karoline's Cakes.



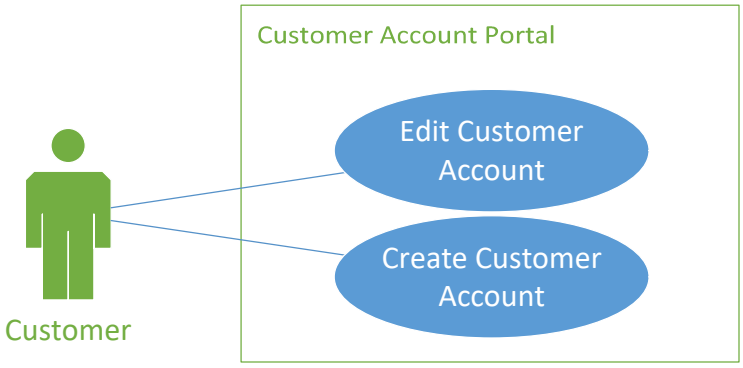
USECASE DIAGRAM: Confirm Weekly Schedule

Management is the Actor. Management will be able to determine their availability inside the scheduling system as well as produce the entire schedule for the week.



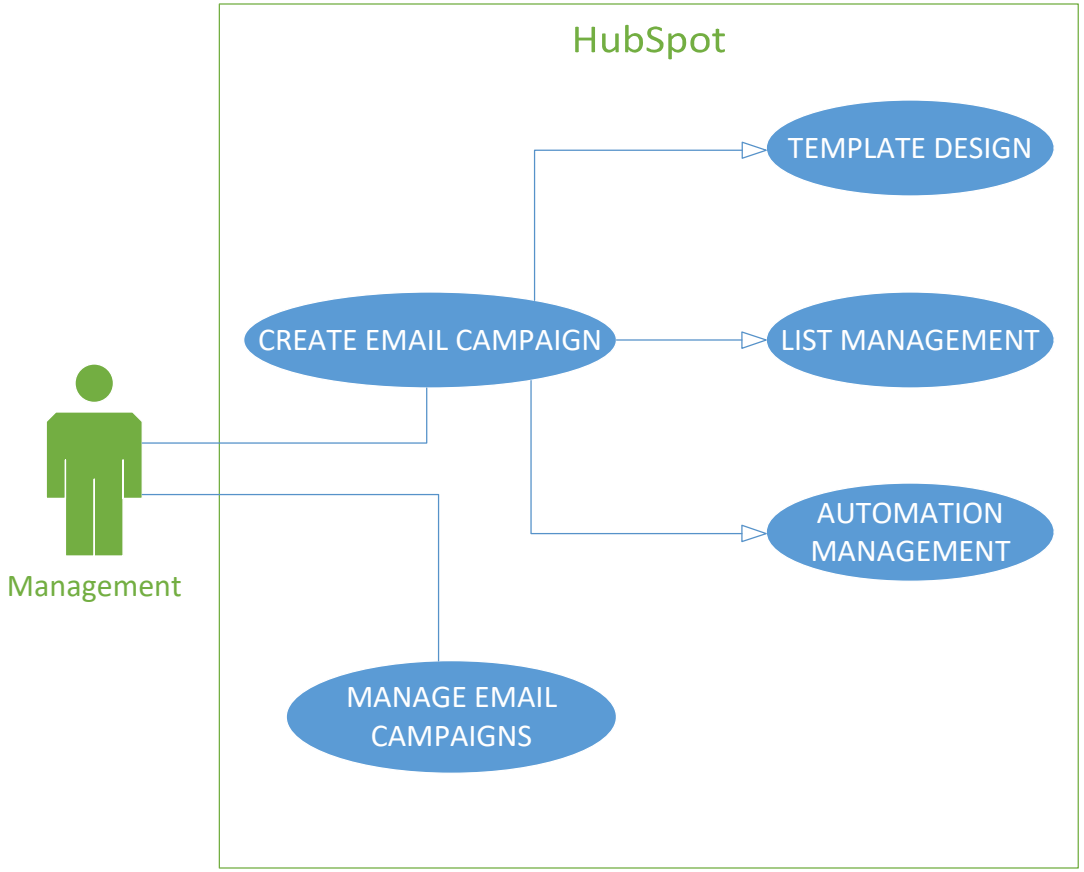
Customer Account

The customer actor is the only actor that edits and creates customer accounts.



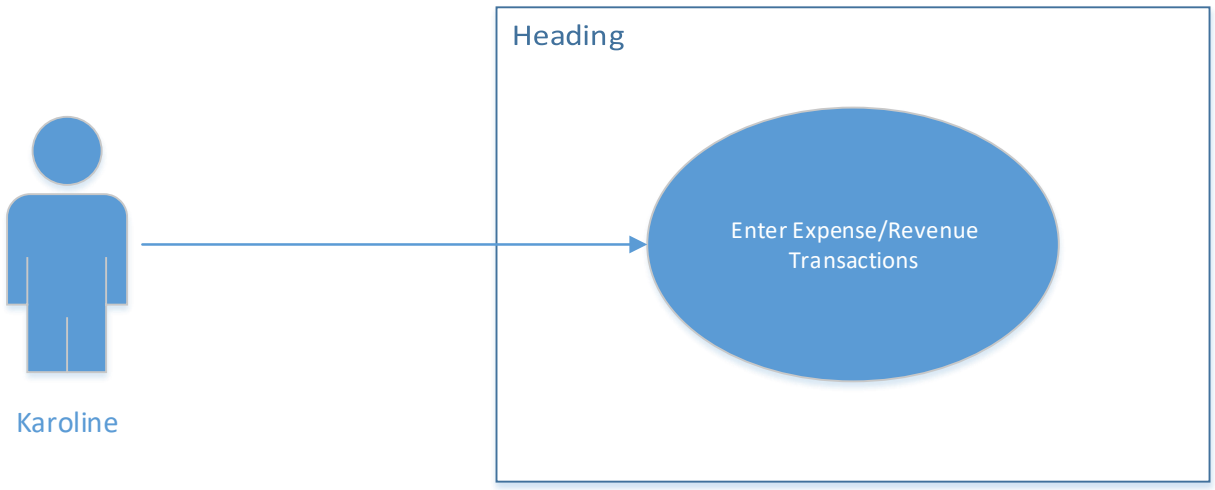
USECASE DIAGRAM: Create Email Campaign

Management is the Actor. Management needs the ability to create individual email campaigns. These campaigns will incorporate templates, use list management to organize email list, and automation to determine the frequency in which emails will be sent.
Management can also manage saved campaigns (e.g. birthday reminders, welcome emails) so they can update the formatting, picture(s), and email list



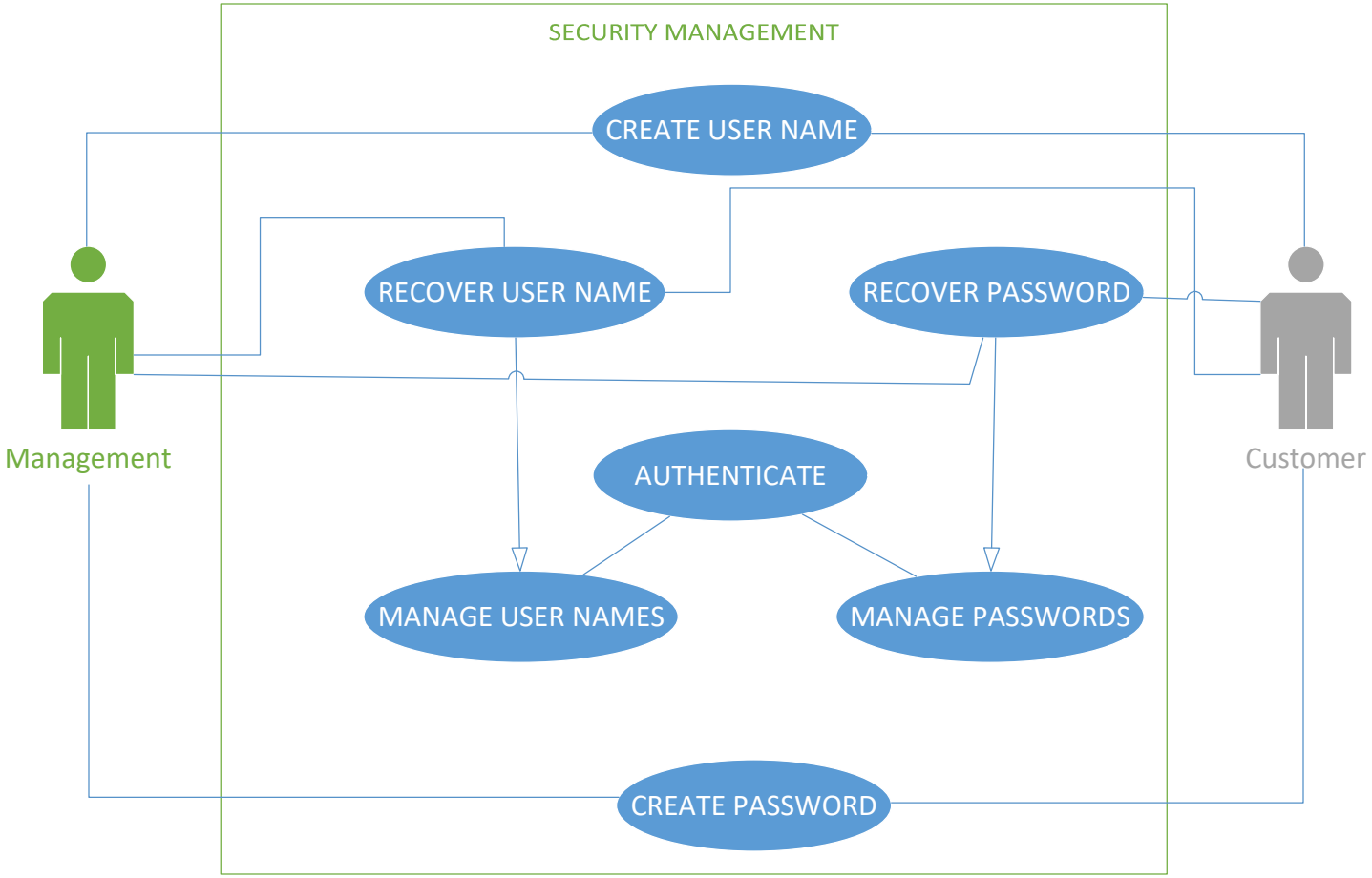
Use Case Diagram: Enter Transactions

This diagram shows Karoline interacting with the system to enter her expense and revenue transactions.



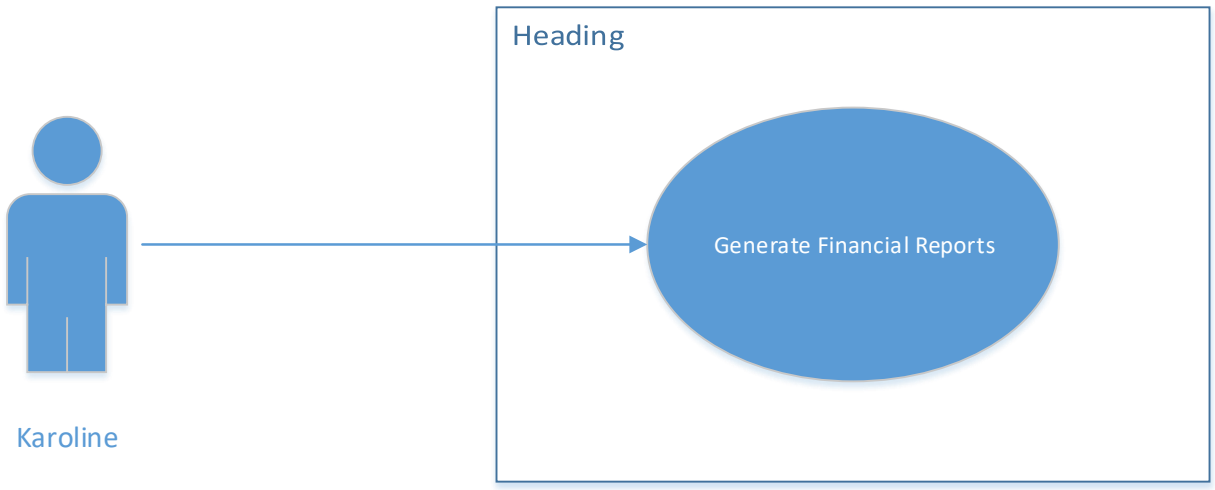
USECASE DIAGRAM: Forgot User Login Info

Management and Customer are the Actors. Both actors need the option of recovering forgotten username and/or password.



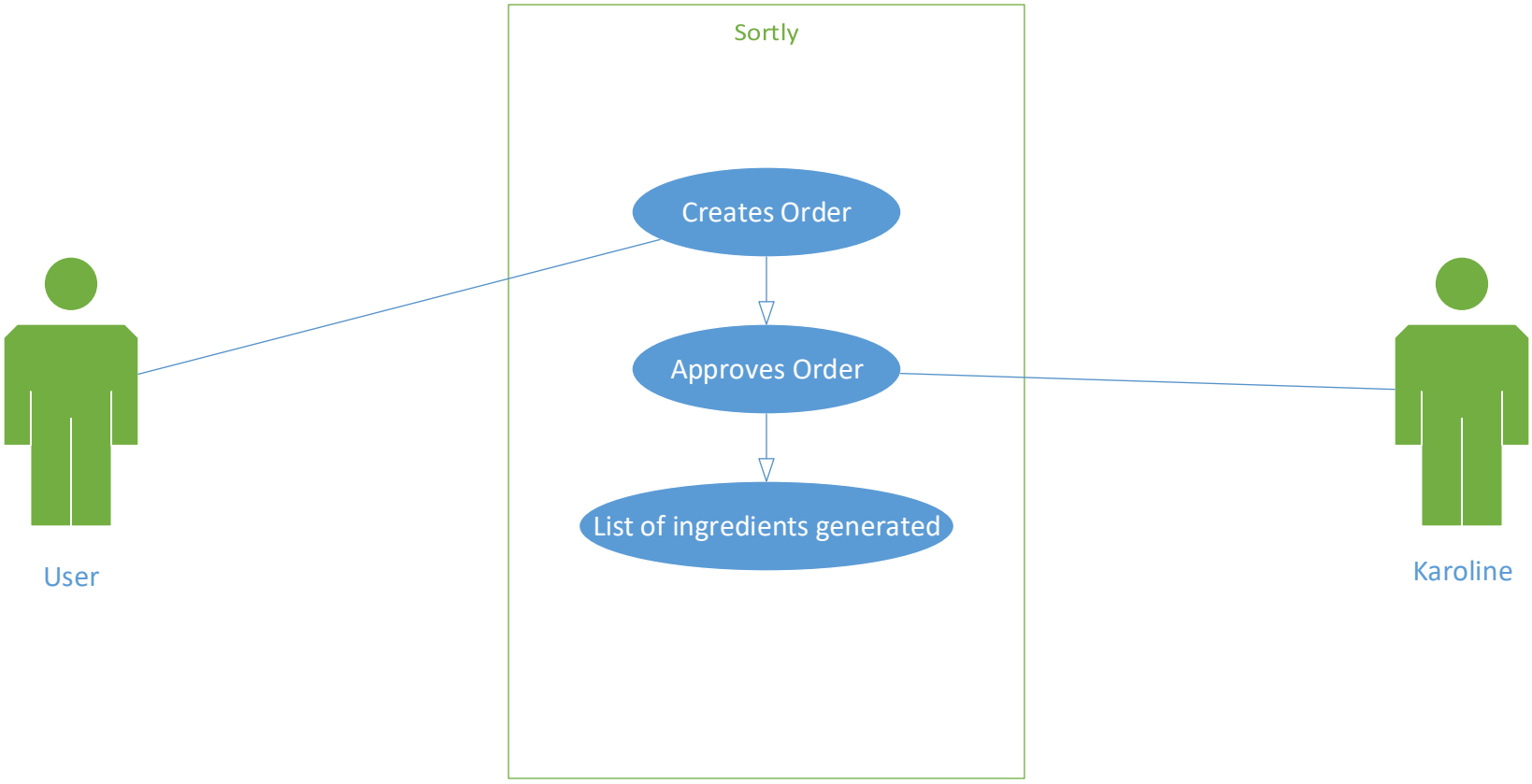
Use Case Diagram: Generate Financial Reports

This diagram shows Karoline interacting with the QuickBooks system to generate Financial Reports.



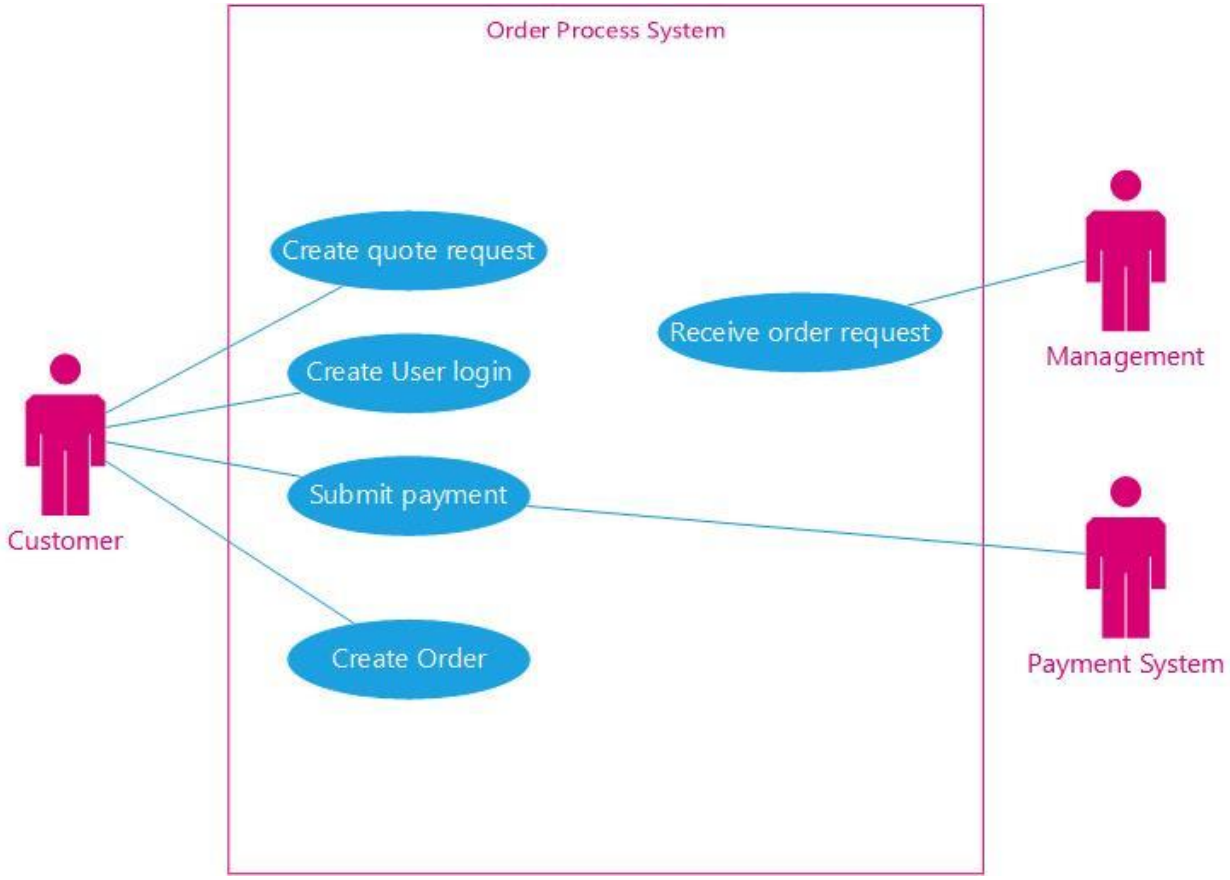
Generate List of Ingredients

The actors in this Use Case are the user and Karoline. The user will create an order, Karoline will approve the order, and an automated list of ingredients will be generated based on what the user orders.



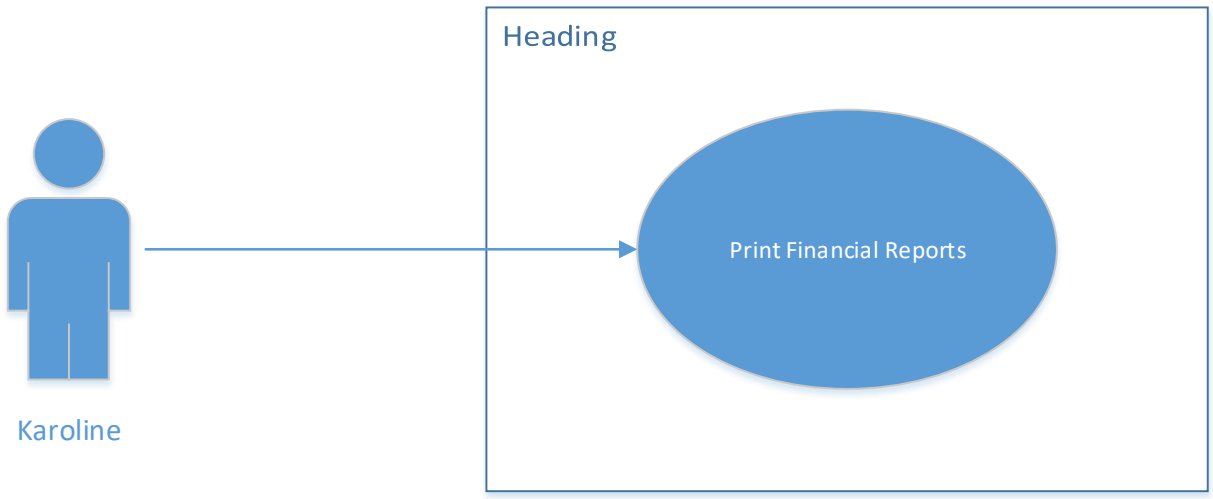
Order process System

The following use case diagram is the order process system. It shows the customer interacts with the high risk use cases create quote, create user login, submit payment, and create order, which leads to an increase in revenue. The low risk use case associated with this process, receive order, does not interact with the customer, but would be used to make the process easier and reduce costs.



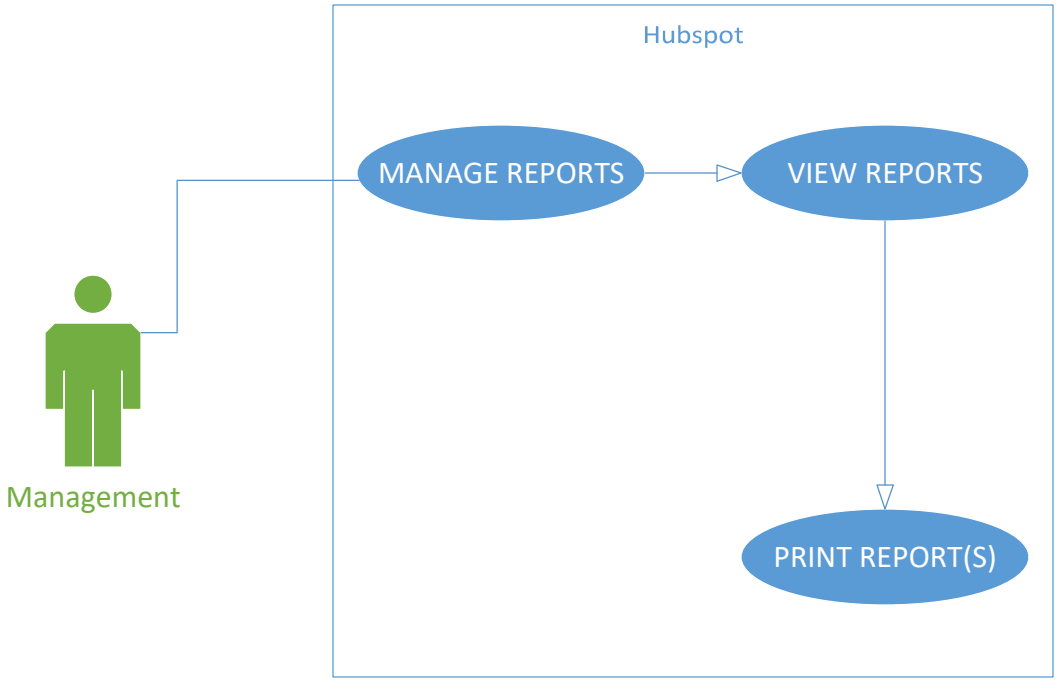
Use Case Diagram: Print Financial Statements

This diagram shows Karoline interacting with the system to print the financial reports.



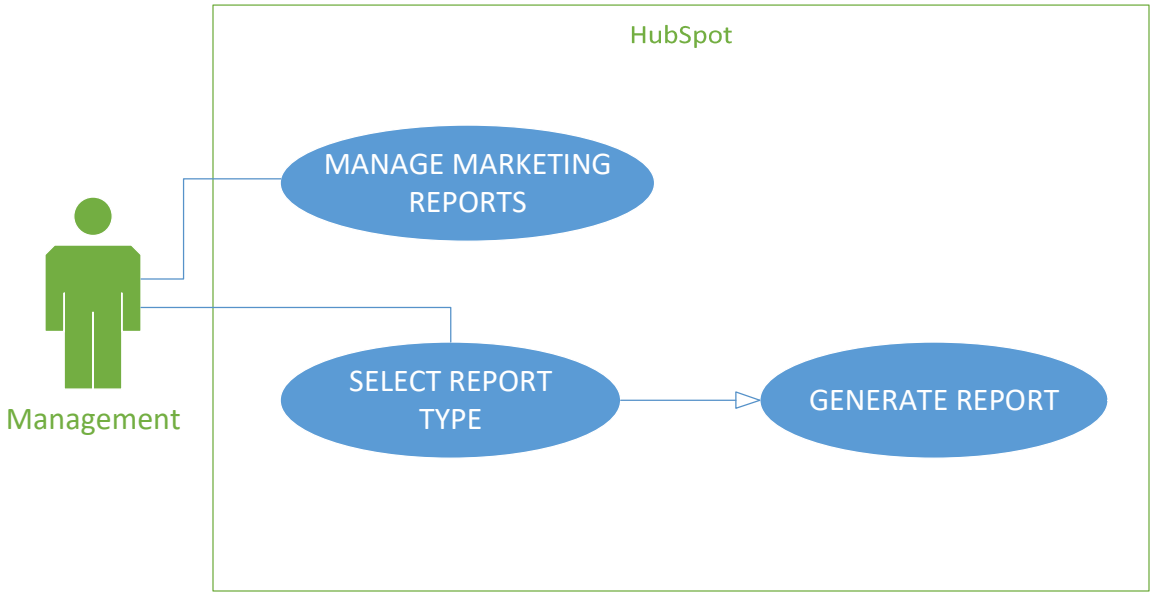
USECASE DIAGRAM: Print Marketing Reports

Management is the Actor. Management needs the ability to print created reports to maintain hard-copy records.



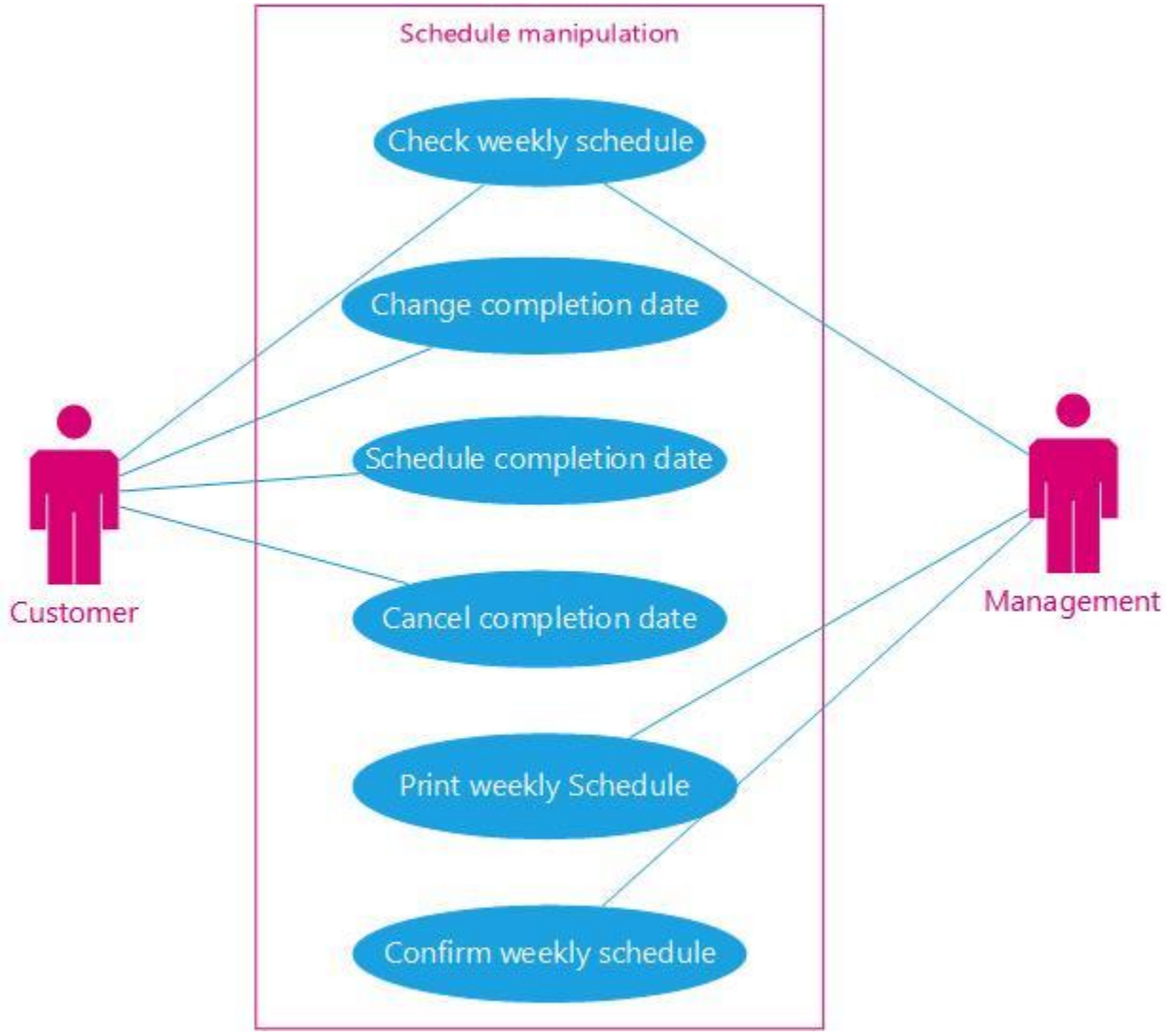
USECASE DIAGRAM: Produce Marketing Reports

Management is the Actor. Management will be able to generate Marketing Reports by selecting all, some, or individual reports from a pre-generated list of reports



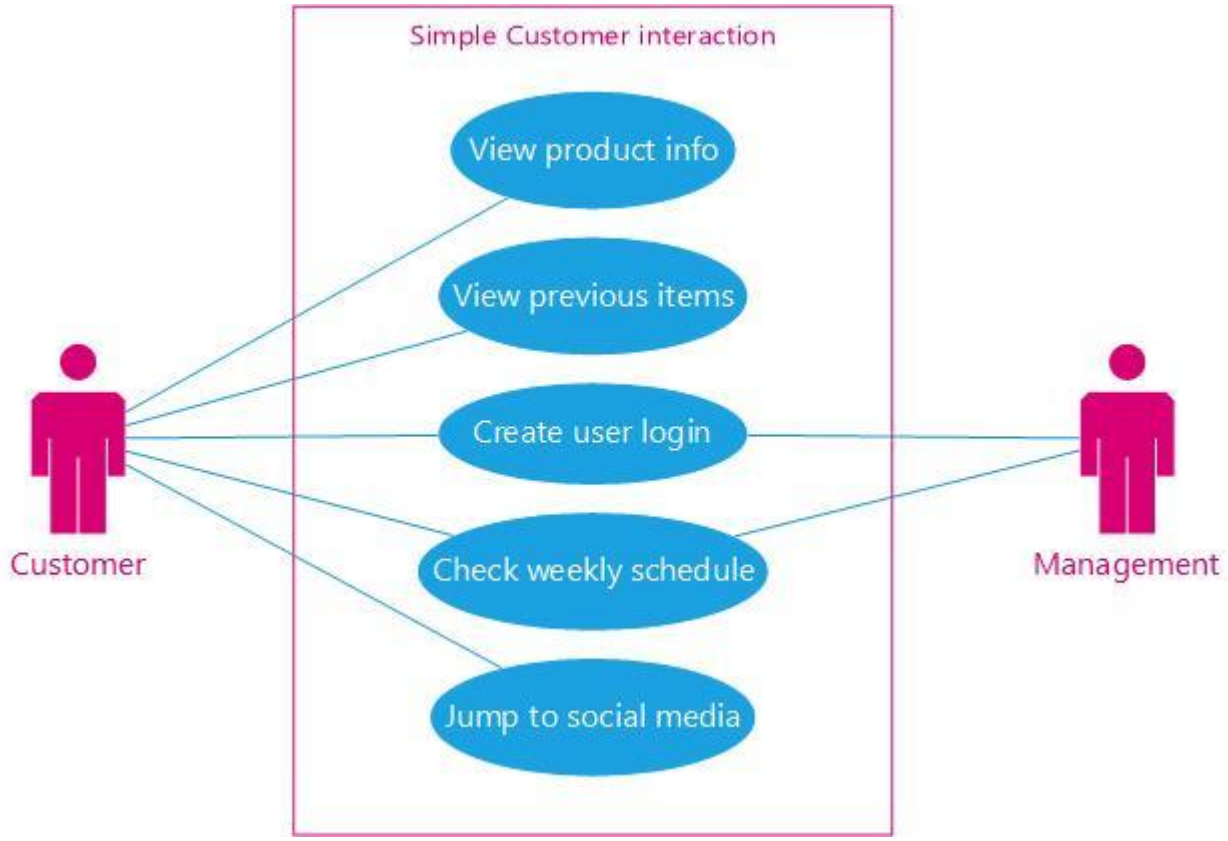
Schedule manipulation

The following use case diagram is for schedule manipulation. It shows the customer interacts directly with all the high risk use cases which are part of the purchasing process. The Management interacts directly with the other low risk use cases. The low risk use cases here do not directly impact revenue, but help in the management process which would lower costs.



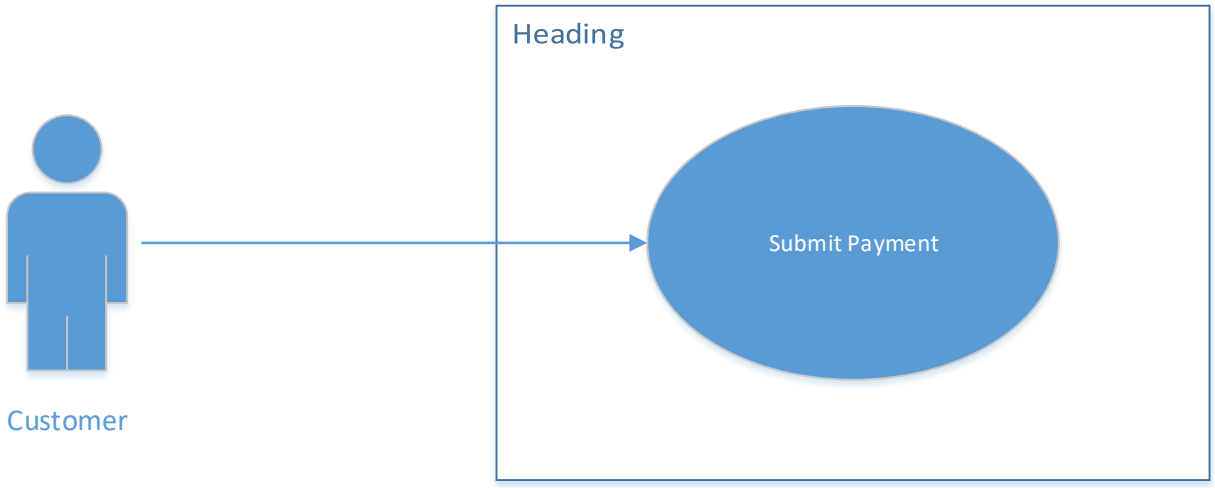
Simple customer interaction

The following use case diagram is for simple customer interaction. It shows the customer interacts with the all the use cases but the jump to social media use case does not directly lead to a change in revenue. The other four use cases all lead to a change in revenue due to them being a part of the customer purchasing process. Sometimes the customer will also interact with low risk use cases.



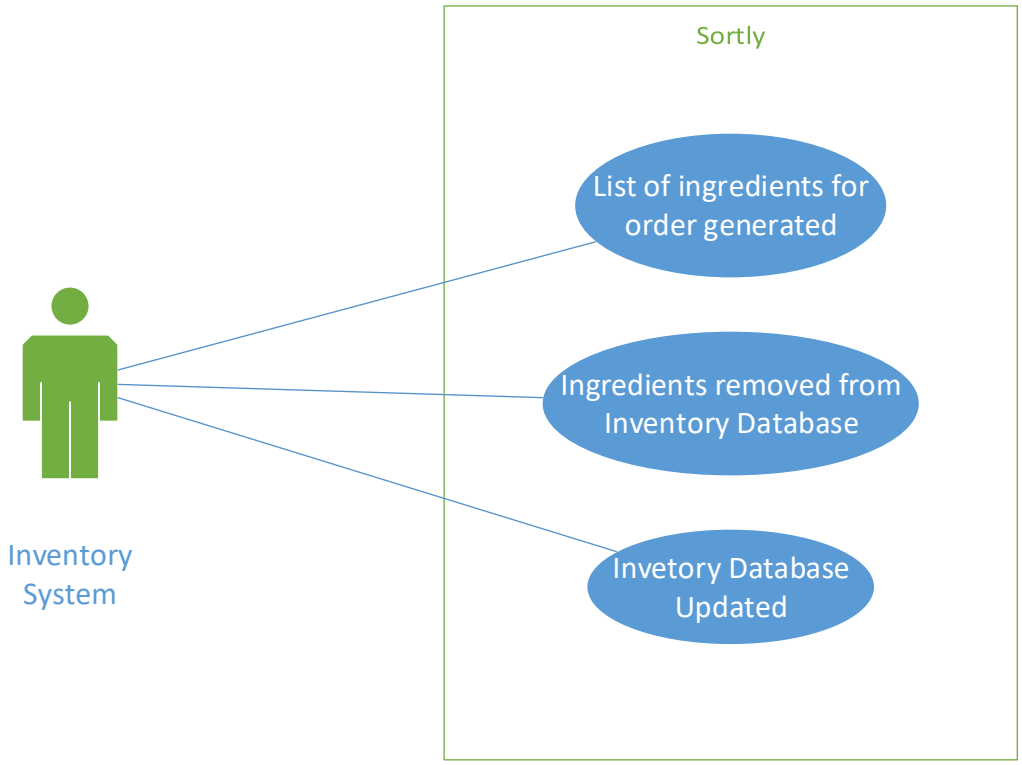
Use Case Diagram: Submit Payment

This diagram shows the user, in this case the customer, interacting with the system to submit a payment.



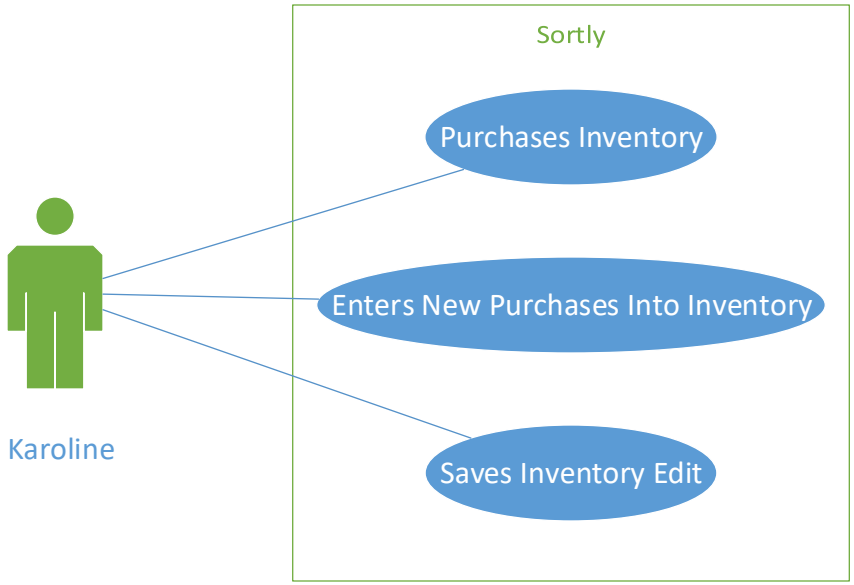
Track Inventory

The actor is the inventory system. The list of ingredients for the order is generated. The ingredients for the order are automatically removed from the inventory database. Then the inventory database is automatically updated.



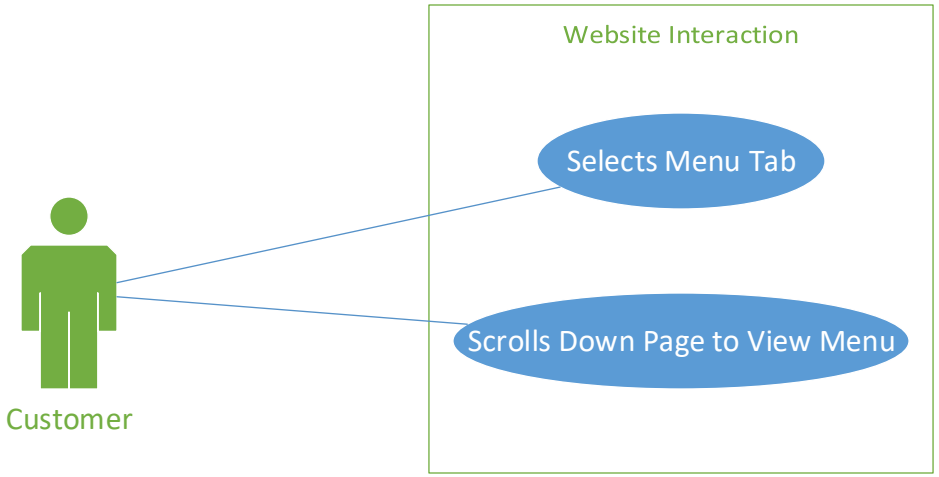
Update Inventory

The actor is Karoline. Karoline purchases inventory, then enters the new ingredients into inventory. After, Karoline saves the inventory edit.



View Product Information

The actor is the user. They select the Menu Tab on Sweet Karoline's Cakes webpage. They scroll down to view the menu.



Use Case Specification: Cancel Completion Date

1. Cancel Completion Date

1.1 Brief Description

This use case describes how a customer would go about cancelling their cake order.

2. Flow of Events

2.1 Basic Flow

1. Customer decides to cancel their order.
2. Customer goes to Sweet Karoline's Cakes website to obtain Karoline's email or phone number.
3. Customer notifies Karoline Gardner through email or by phone.
4. Karoline logs into scheduling system.
5. Karoline locates the appointment on her list.
6. Karoline clicks the "Mark As" drop down box.
7. Karoline marks the appointment as cancelled.
8. Karoline clicks on the "Cancelled" appointments tab.
9. Karoline deletes the appointment from the list.

3. Pre-conditions

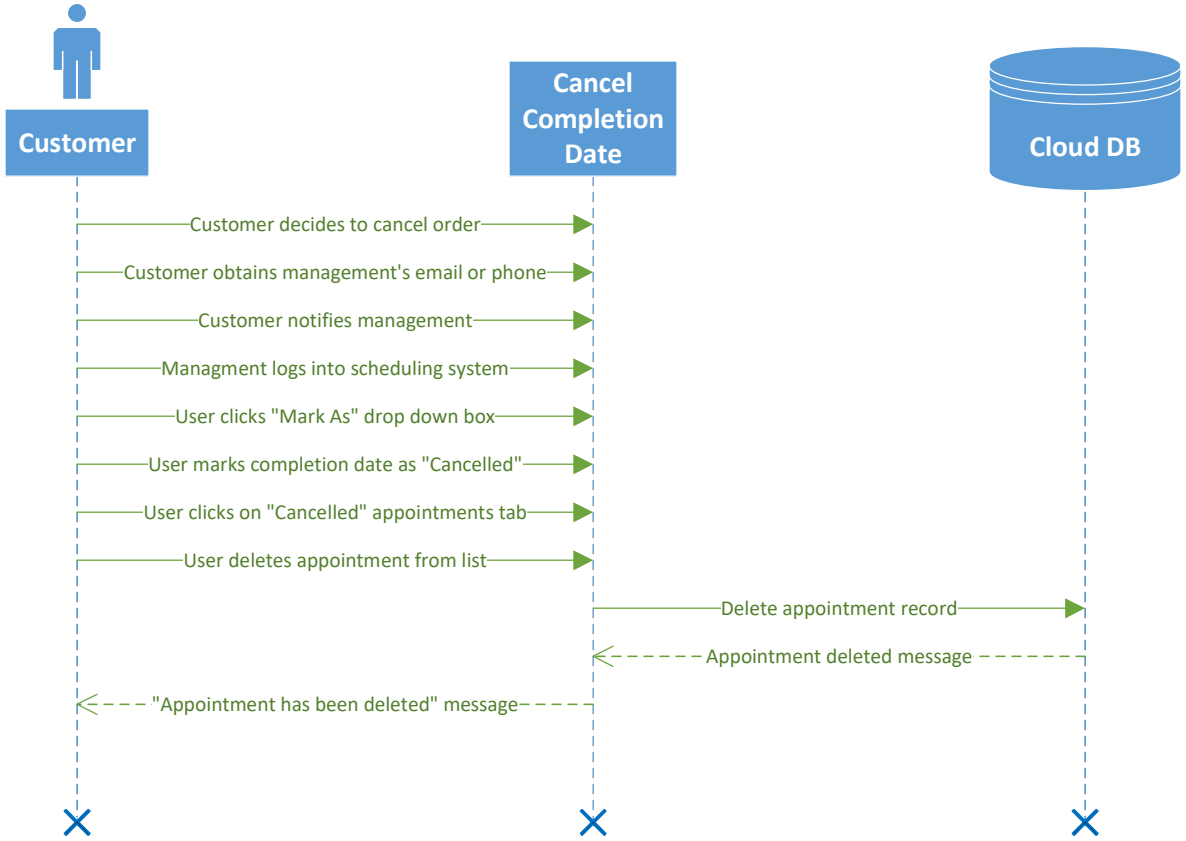
3.1 Customer has completed a quote request form.

3.2 Customer has scheduled an appointment date and time.

3.3 Karoline has approved of the original requested date and time of completion for the order.

Cancel Completion Date

1. Customer decides to cancel their order.
2. Customer goes to Sweet Karoline's Cakes website to obtain Karoline's email or phone number
3. Customer notifies Karoline Gardner through email or by phone.
4. Karoline logs into scheduling system.
5. Karoline locates the appointment on her list.
6. Karoline clicks the "Mark As" drop down box.
7. Karoline marks the appointment as cancelled.
8. Karoline clicks on the "Cancelled" appointments tab.
9. Karoline deletes the appointment from the list.



Use Case Specification: Change Completion Date

1. Change Completion Date

1.1 Brief Description

This use case describes how a customer would go about changing the date of completion for the cake they are ordering.

2. Flow of Events

2.1 Basic Flow

1. Customer decides they want to change the completion date of the cake they ordered.
2. Customer visits website to observe other available times on the calendar widget.
3. Customer chooses desired date/time they want to change the completion date to.
4. Customer notifies Karoline Gardner of the desired change by phone or email.
5. Karoline goes into scheduling system software on her laptop
6. Karoline clicks “Reschedule” to edit the customer’s appointment.
7. Karoline replaces the current date and time on the appointment with the revised date and time.
8. Scheduling system sends a receipt to the customer notifying them of the completed change to their order time and date.

2.2 Alternative Flows

2.2.1 Karoline denies customer’s desired date/time

1. Karoline notifies customer of the denial and allows them to choose another date/time.
2. Customer chooses new date/time for their order.
3. Customer notifies Karoline of the new date/time they desire.
4. Karoline approves of the new appointment and repeats the “Reschedule” process from basic flow.

3. Pre-conditions

3.1 Customer has completed a quote request form.

3.2 Customer has scheduled an appointment date and time.

3.3 Karoline has approved of the original requested date and time of completion for the order.

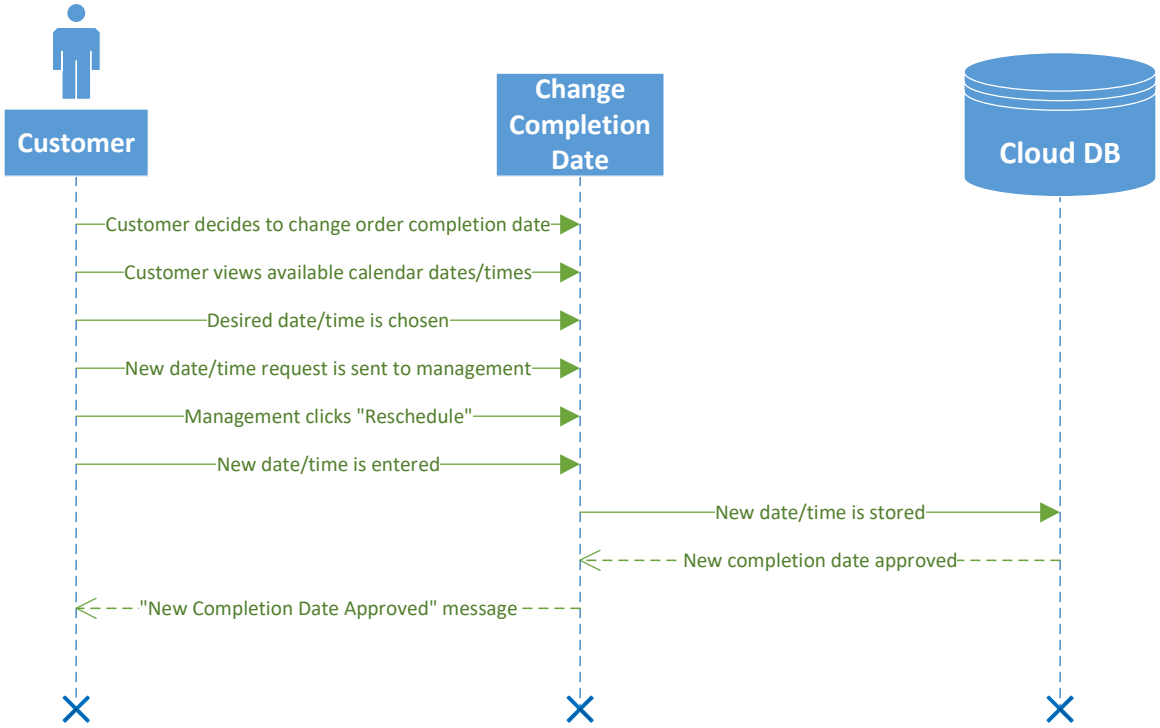
4. Post-conditions

4.1 Completion date and time are updated in the scheduling system.

4.2 Receipt is sent to customer confirming new completion date.

Change Completion Date

1. Customer decides they want to change the completion date of the cake they ordered.
2. Customer visits website to observe other available times on the calendar widget.
3. Customer chooses desired date/time they want to change the completion date to.
4. Customer notifies Karoline Gardner of the desired change by phone or email.
5. Karoline goes into scheduling system software on her laptop
6. Karoline clicks “Reschedule” to edit the customer’s appointment.
7. Karoline replaces the current date and time on the appointment with the revised date and time.
8. Scheduling system sends a receipt to the customer notifying them of the completed change to their order time and date.
9. Scheduling system alerts Karoline Gardner of a new appointment being made.



Use Case Specification: Check Weekly Schedule

1. Check Weekly Schedule

1.1 Brief Description

This use case describes how the user would check the schedule of cake deadlines for the current week.

2. Flow of Events

2.1 Basic Flow

1. User decides to look at the current availability of dates/times for cake appointments in the current week.
2. User goes to Sweet Karoline’s Cakes website.
3. User clicks on calendar widget on Sweet Karoline’s Cakes website homepage.
4. Scheduling system shows blacked out as well as open dates/times for cake appointments during the current week.

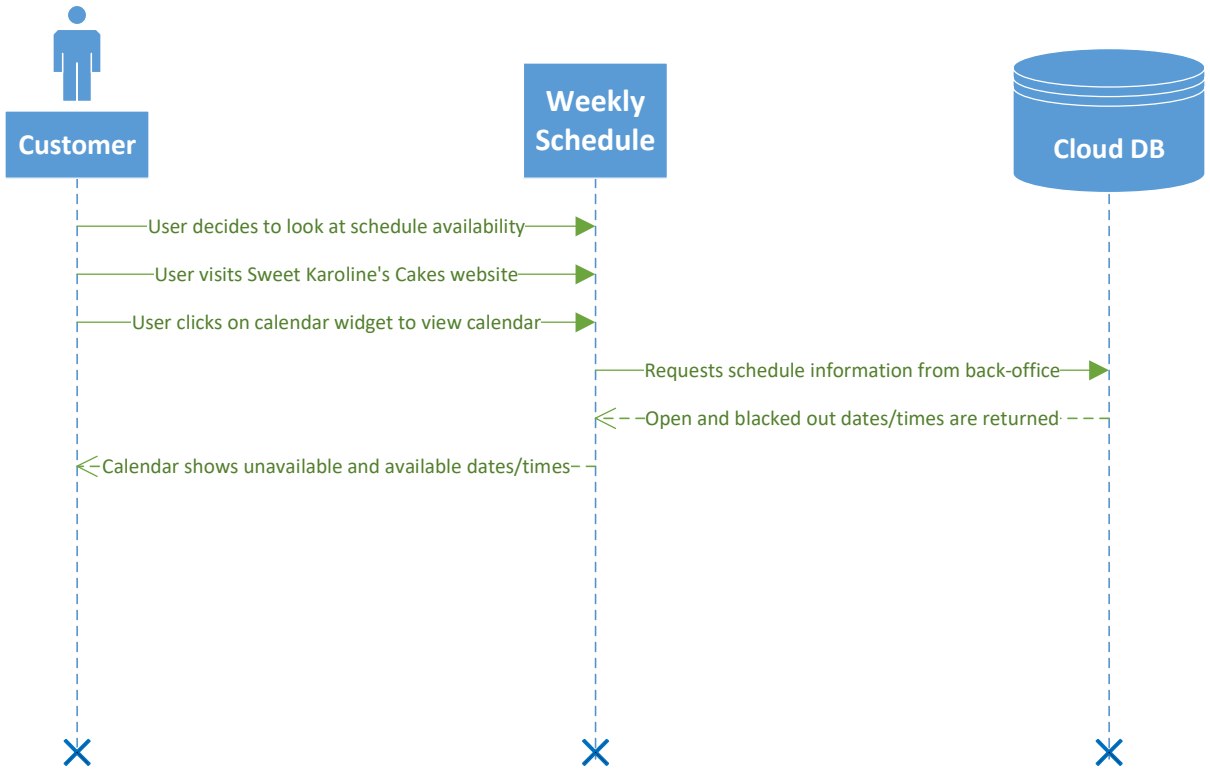
3. Pre-conditions

3.1 Order request form must be submitted.

3.2 Appointments must be approved by Karoline to show up on the calendar.

Check Weekly Schedule

1. User decides to look at the current availability of dates/times for cake appointments in the current week.
2. User goes to Sweet Karoline’s Cakes website.
3. User clicks on calendar widget on Sweet Karoline’s Cakes website homepage.
4. Scheduling system shows blacked out as well as open dates/times for cake appointments during the current week.



Use Case Specification: Confirm Weekly Schedule

1.1 Brief Description

Karoline will confirm feasibility of weekly schedule to ensure deadlines are both scheduled and kept.

2. Flow of Events

2.1 Basic Flow

Actor(s): Karoline

Karoline selects a day

Karoline approves customer submitted event (appointment, pick-up, delivery, etc.)

Karoline adds any management events to calendar

Karoline edits customer/management events to calendar to reflect appropriate changes

Karoline saves schedule

2.2 Alternative Flows

2.2.1 Karoline may DELETE events from the calendar

Actor(s): Karoline

Karoline deletes customer event(s)

System emails customer(s) to confirm cancelation of event

3. Special Requirements

3.1 Schedule ADD, DELETE, EDIT restrictions

For security reasons and to resolve scheduling conflicts, ONLY Karoline will have access to ADD, DELETE, or EDIT the weekly schedule.

4. Pre-conditions

4.1 It is the beginning of Karoline's established work week.

Schedule will be reviewed at the beginning of the week to resolve any conflicts and to reflect changes.

4.2 Customer(s) schedule appointments, deliveries, and/or pickups

If no customer events are scheduled, the schedule will reflect this as *free time*
If customers requested events are approved by Karoline, they will be detailed on the weekly schedule.

4.3 Karoline schedules appointments, deliveries, and/or pickups

Karoline may use the schedule to reflect times where the business will not operate, or block out time for business related activities (i.e. shopping, lengthy deliveries, scheduled maintenance, etc.).

5. Post-conditions

5.1 Weekly Schedule will be saved and printed

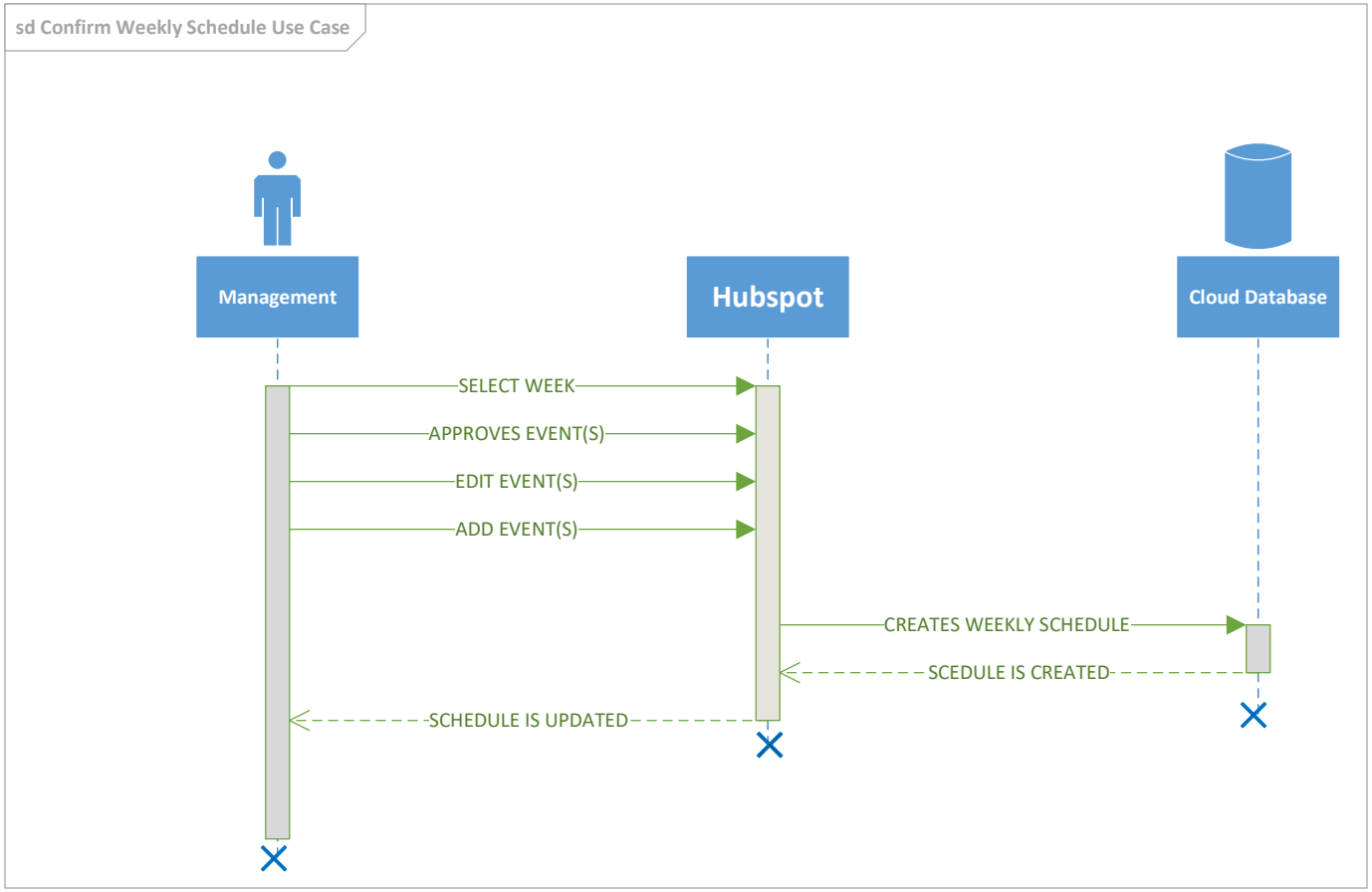
6. Extension Points

6.1 No Extension Point

SEQUENCE DIAGRAM: Confirm Weekly Schedule

MAINFLOW

1. Management selects a day
2. Management approves customer submitted event (appointment, pick-up, delivery, etc.)
3. Management adds any management events to calendar
4. Management edits customer/management events to calendar to reflect appropriate changes
5. Management saves schedule



Use Case Specification: Create Customer Account Information

1. Create Customer Account Information

1.1 Brief Description

When a customer fills out an account creation form their information will be stored for later retrieval and allow for account management.

2. Flow of Events

2.1 Basic Flow

The following fields are input by the customer in the Create Account Form. The customer inputs their first name. The customer inputs their last name. The customer inputs their address line1. The customer inputs their address line2. The customer inputs their city. The customer inputs their state. The customer inputs their zip-code. The customer inputs their email. The customer inputs their phone number. The customer inputs their desired username. The customer inputs their desired password.

The Create Account Form makes a PUT call to the cloud database. The database checks if the desired username is a duplicate and that all fields are valid. The database responds with a success or failure message to the Create Account Form. The Create Account Form relays the database response to the customer.

2.2 Alternative Flows

3. Special Requirements

3.1 Digital Security of Personally Identifiable Information

Extra steps for digital security should be taken to protect the PII. Encryption of data is suggested.

4. Pre-conditions

4.1 Key Customer Inputs Are Unique

Some fields for a customer record need to be unique. Emails shall not be duplicated. Name and Address combination shall not be duplicated. Desired username shall not already exist.

4.2 Form Inputs are Valid

Form inputs must be valid base on field. Phone numbers must be in the correct format. Email must be a valid email. Zip must be in the correct format. City must be a valid city. State must be valid state.

5. Post-conditions

5.1 The Customer’s Account is Created

The customer’s account is created and stored in the database. Now the account information is accessible via logging in.

6. Extension Points

6.1 Customer Account Information Already Exists

The following fields are input by the customer in the Create Account Form. The customer inputs their first name. The customer inputs their last name. The customer inputs their address line1. The customer inputs their address line2. The customer inputs their city. The customer inputs their state. The customer inputs their zip-code. The customer inputs their email. The customer inputs their phone number. The customer inputs their desired username. The customer inputs their desired password.

The Create Account Form makes a PUT call to the cloud database. The database checks if the desired username is a duplicate and that all fields are valid. If the Customer Account already exists, the database responds with a failure message to the Create Account Form. No data is stored yet.

The customer will receive the message on the Create Account Form page and asked to edit the duplicate fields or sign in to the pre-existing account.

6.2 Desired Username Already Exists

The following fields are input by the customer in the Create Account Form. The customer inputs their first name. The customer inputs their last name. The customer inputs their address line1. The customer inputs their address line2. The customer inputs their city. The customer inputs their state. The customer inputs their zip-code. The customer inputs their email. The customer inputs their phone number. The customer inputs their desired username. The customer inputs their desired password.

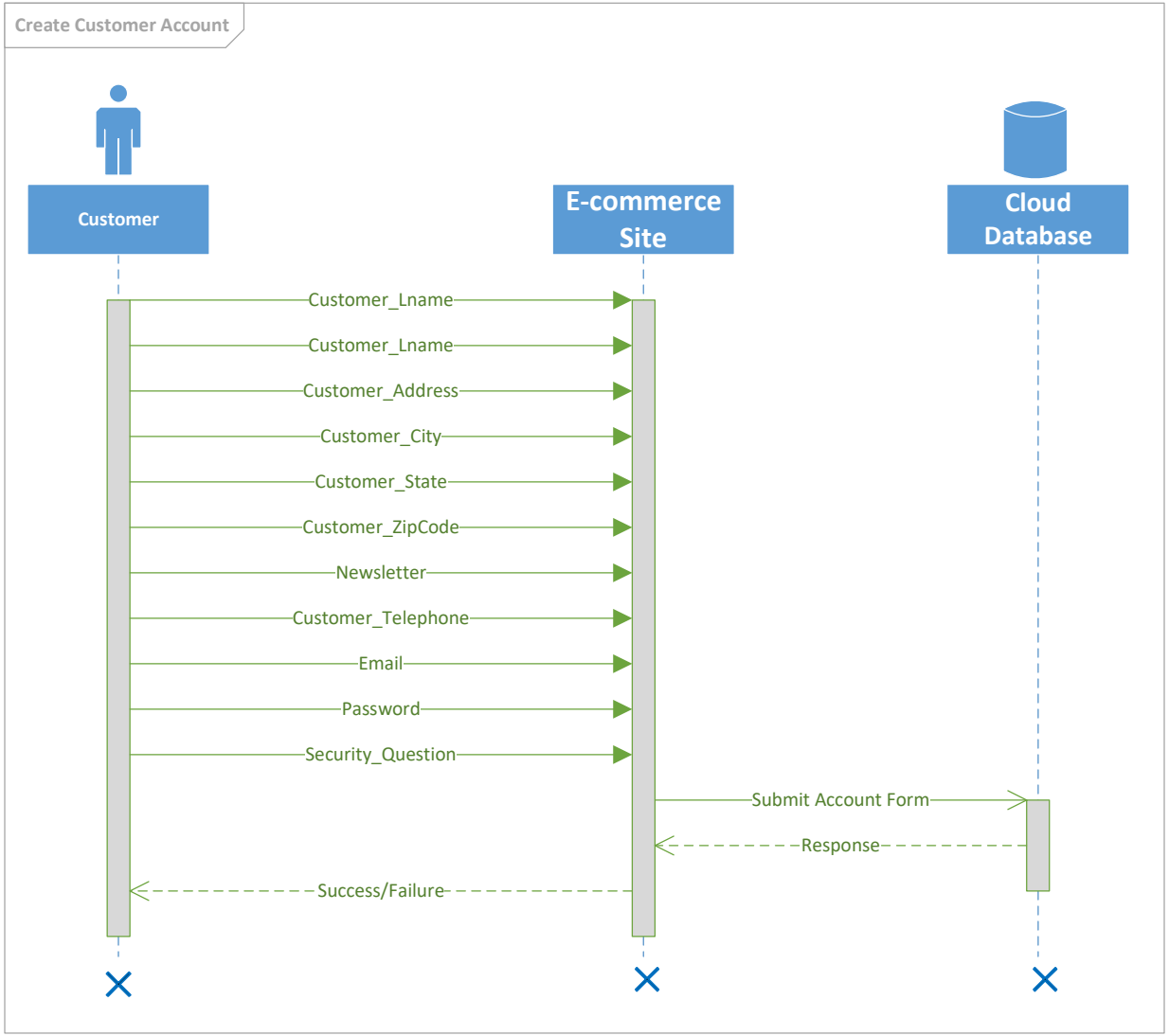
The Create Account Form makes a PUT call to the cloud database. The database checks if the desired username is a duplicate and that all fields are valid. If the desired username already exists, the database responds with a failure message to the Create Account Form. No data is stored yet.

The customer will receive the message on the Create Account Form page and asked to change the duplicate username or sign in to the pre-existing account.

SEQUENCE DIAGRAM: Create Customer Account

MAIN FLOW

- The following fields are input by the customer in the Create Account Form.
- The customer inputs their first name.
- The customer inputs their last name.
- The customer inputs their address line1.
- The customer inputs their address line2.
- The customer inputs their city.
- The customer inputs their state.
- The customer inputs their zip-code.
- The customer inputs their email.
- The customer inputs their phone number.
- The customer inputs their desired username.
- The customer inputs their desired password.
- The Create Account Form makes a PUT call to the cloud database. The database checks if the desired username is a duplicate and that all fields are valid.
- The database responds with a success or failure message to the Create Account Form.
- The Create Account Form relays the database response to the customer.



Use Case Specification: Create Email Campaign

1. Create Email Campaign

1.1 Brief Description

Karoline will generate email marketing material by utilizing CRM software to establish business centric campaigns (i.e. wedding specials, birthday reminders, etc.) on either premade or custom templates.

2. Flow of Events

2.1 Basic Flow

Actors: Karoline, CRM software

Karoline will use a premade email template to distribute information, sales, and other material to customers. System will display selected template for EDITING, PRINTING, DELETING, and SENDING.

2.1.1 Image is uploaded

If image(s) will be used in the email marketing campaign, a .jpeg will be uploaded by Karoline. This image will then be edited to ensure cohesiveness and formatting to ensure proper display.

2.2 Alternative Flows

2.2.1 Custom template is selected

Actors: Karoline, CRM software

Karoline may select a custom template design and structure the layout as she sees fit.

2.2.2 Email campaign is AUTOMATICALLY distributed

Actors: Karoline, CRM software

Karoline may select particular email campaigns to reoccur at a particular date/anniversary

This feature will include any/all recipients that Karoline wants and can be edited before the distribution date.

3. Special Requirements

There are no Special Requirements

4. Pre-conditions

4.1 Access to CRM software for customer email list is required

Karoline must have access/permission to login to CRM software to CREATE/EDIT/DELETE email and/or email list.

4.2 Customer email(s) must be stored using the CRM software.

New and existing customer's information that is obtained must be entered into the CRM software program.

5. Post-conditions

5.1 Created email campaigns will be sent

Email Campaigns that are not set to send automatically MUST be sent by Karoline at the appropriate time.

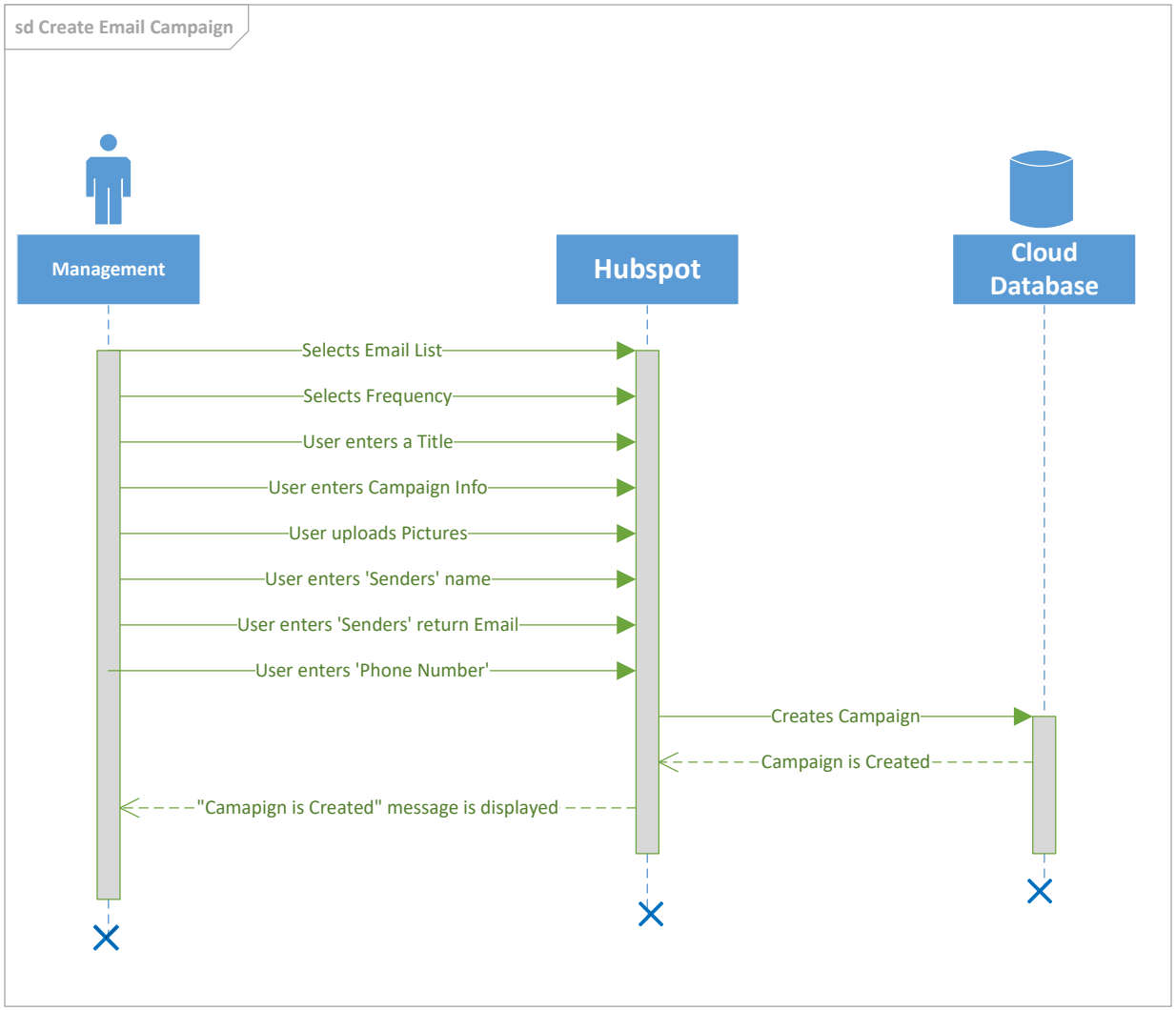
6. Extension Points

There are no Extension Points

SEQUENCE DIAGRAM: Create Email Campaign

MAIN FLOW

1. User selects email list from list box or paste email list into list box
2. User selects email frequency from automation drop down box
3. User types email campaign title in title textbox
4. User types or paste campaign information into the body textbox
5. User clicks on '+ image/video' button to upload images or videos
6. User types senders name in Name textbox
7. User enters senders return email in Email textbox
8. Users enters phone number in Phone textbox
9. User clicks '+ email campaign' button
10. System saves campaign



Use Case Specification: Create order

1. Create order

1.1 Brief Description

A customer will be able to create an order request for non-custom products listed on the website's product information page.

2. Flow of Events

2.1 Basic Flow

Actors: Customer

User enters first name.

User enters last name.

User enters telephone number.

User enters email address.

User enters product.

User enters quantity.

User enters completion date.

User selects pickup or delivery.

If delivery User enters street address.

If delivery User enters city.

If delivery User enters state.

If delivery User enters zip code.

User enters clicks submit button.

2.2 Alternative Flow

2.2.1 User is logged into account

Actors: Customer

User enters product.

User enters quantity.

User enters completion date.

User selects pickup or delivery.

User enters clicks submit

3. Pre-conditions

3.1 User selected place order

Before an order can be created the user must be have selected place order.

4. Post-conditions

4.1 User sent to payment processor

After the order is created the user will be sent to a third-party payment processing webpage to complete the payment process.

4.2 Order request is sent to Karoline

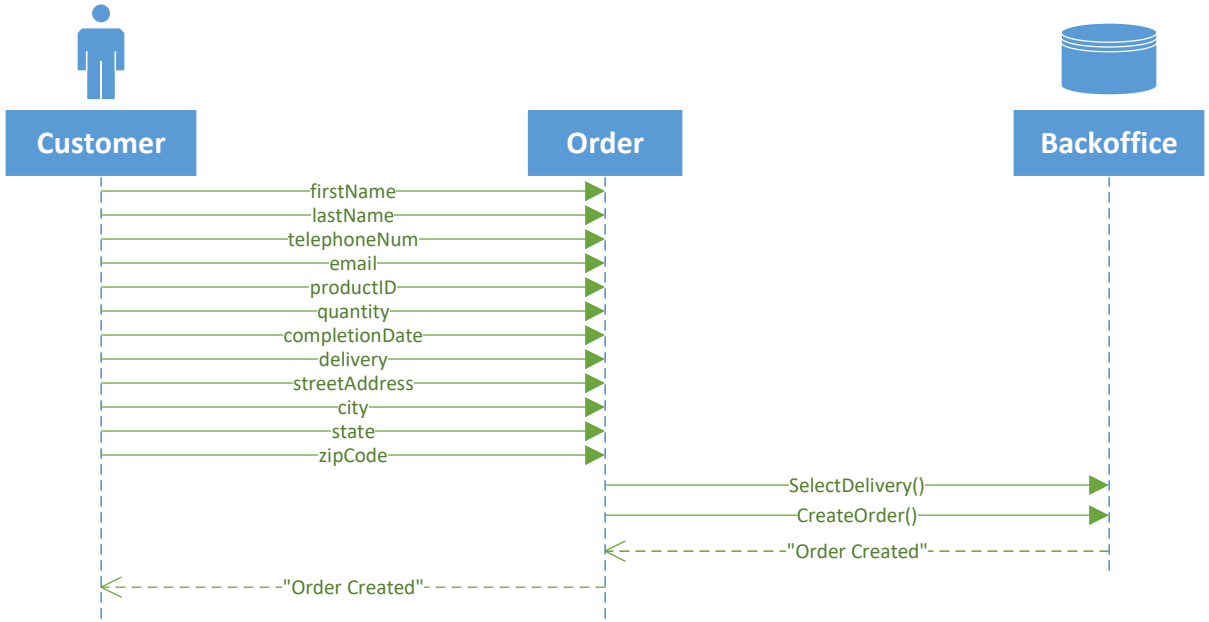
After the order is created an email will be generated and sent to Karoline containing all the information.

4.3 Order information is sent to back office

After the order is created the information will sent to the back-office software.

Main flow of events:

1. Customer enters information.
2. Customer selects Delivery option.
3. Customer clicks “Submit” button.
4. System creates order.
5. System returns “order created” notification.



Use Case Specification: Create quote request

1. Create quote request

1.1 Brief Description

A customer will be able to create a quote request for custom made products.

2. Flow of Events

2.1 Basic Flow

Actors: Customer

User enters first name.

User enters last name.

User enters telephone number.

User enters email address.

User enters serving size.

User enters tier count.

User enters budget.

User enters cake flavors.

User enters cake filling.

User enters frosting flavors

User enters any addition comments.

User may upload an image.

User enters completion date.

User selects pickup or delivery.

If delivery User enters street address.

If delivery User enters city.

If delivery User enters state.

If delivery User enters zip code.

User enters clicks submit button.

2.2 Alternative Flow

2.2.1 User is logged into account

Actors: Customer

User enters serving size.

User enters tier count.

User enters budget.

User enters cake flavors.

User enters cake filling.

User enters frosting flavors

User enters any addition comments.

User enters completion date.

User may upload an image.

User selects pickup or delivery.

2.2.2 User uploads image.

Actors: Customer

User selects browse for image.

User selects image for upload

User clicks submit.

3. Pre-conditions

3.1 User selected quote request

Before a quote request can be created the user must have selected request quote.

4. Post-conditions

4.1 User sent to payment processor

After the order is created the user will be sent to a third-party payment processing webpage to complete the payment process for the down payment.

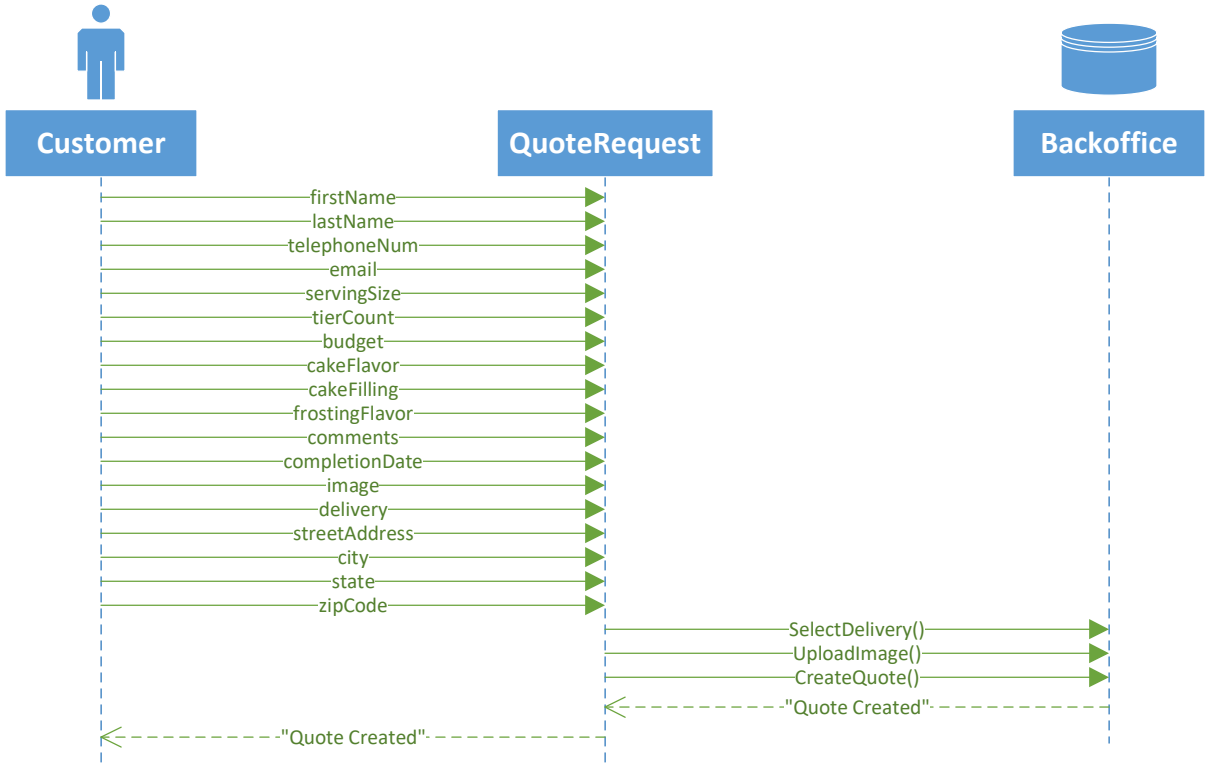
4.2 Quote request is sent to Karoline

After the quote request is created an email will be generated and sent to Karoline containing all the information.

4.3 Order information is sent to back office

After the order is created the information will sent to the back-office software.

Main flow of events:
1. Customer enters information.
2. Customer selects Delivery option.
4. Customer uploads image if desired.
3. Customer clicks “Submit” button.
4. System creates quote.
5. System returns “quote created” notification.



Use Case Specification: Edit Customer Account Information

1. Create Customer Account Information

1.1 Brief Description

Customers should be able to log into their account and edit the account's fields to reflect changes.

2. Flow of Events

2.1 Basic Flow

The customer logs into their account. The customer selects the *edit* button next to the desired field. The customer makes the desired changes to the field and submits the change. The changes are submitted to the database. The database responds with a success message. The customer account page relays the database's message to the customer.

2.2 Alternative Flows

None.

3. Special Requirements

3.1 Digital Security of Personally Identifiable Information

Extra steps for digital security should be taken to protect the PII. Encryption of data is suggested.

4. Pre-conditions

4.1 Form Inputs are Valid

Form inputs must be valid base on field. Phone numbers must be in the correct format. Email must be a valid email. Zip must be in the correct format. City must be a valid city. State must be valid state.

5. Post-conditions

5.1 The Customer's Account is Created

The customer's account is created and stored in the database. Now the account information is accessible via logging in.

6. Extension Points

6.1 Customer Account Information Already Exists

The following fields are input by the customer in the Create Account Form. The customer inputs their first name. The customer inputs their last name. The customer inputs their address line1. The customer inputs their address line2. The customer inputs their city. The customer inputs their state. The customer inputs their zip-code. The customer inputs their email. The customer inputs their phone number. The customer inputs their desired username. The customer inputs their desired password.

The Create Account Form makes a PUT call to the cloud database. The database checks if the desired username is a duplicate and that all fields are valid. If the Customer Account already exists, the database responds with a failure message to the Create Account Form. No data is stored yet.

The customer will receive the message on the Create Account Form page and asked to edit the duplicate fields or sign in to the pre-existing account.

6.2 Desired Username Already Exists

The following fields are input by the customer in the Create Account Form. The customer inputs their first name. The customer inputs their last name. The customer inputs their address line1. The customer inputs their address line2. The customer inputs their city. The customer inputs their state. The customer inputs their zip-code. The customer inputs their email. The customer inputs their phone number. The customer inputs their desired username. The customer inputs their desired password.

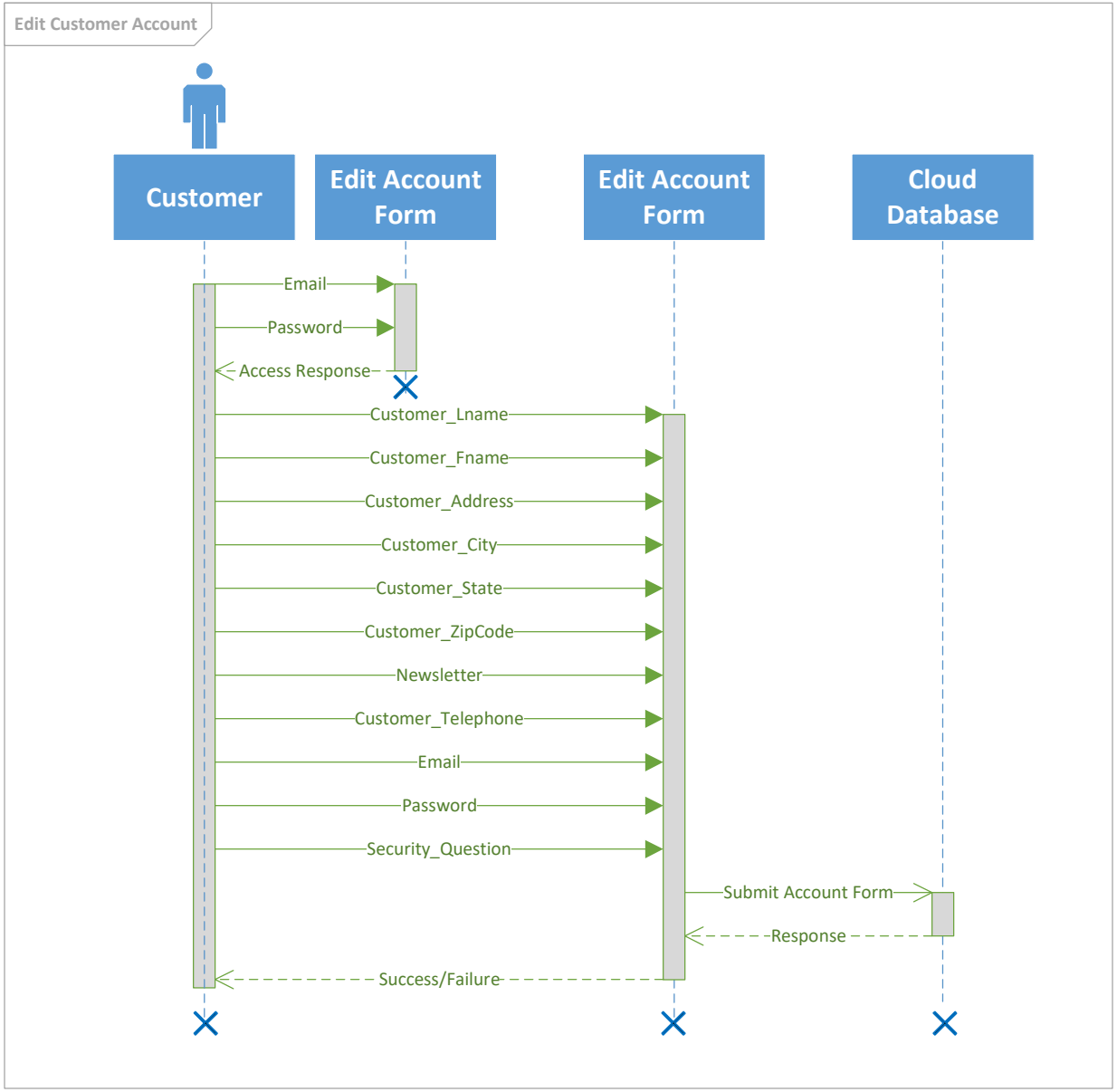
The Create Account Form makes a PUT call to the cloud database. The database checks if the desired username is a duplicate and that all fields are valid. If the desired username already exists, the database responds with a failure message to the Create Account Form. No data is stored yet.

The customer will receive the message on the Create Account Form page and asked to change the duplicate username or sign in to the pre-existing account.

SEQUENCE DIAGRAM: Edit Customer Account

MAIN FLOW

- The customer logs into their account.
- The customer selects the *edit* button next to the desired field.
- The customer makes the desired changes to the field and submits the change.
- The changes are submitted to the database.
- The database responds with a success message.
- The customer account page relays the database's message to the customer.



Use Case Specification: Enter Transactions

1. Use-Case Name

1.1 Brief Description

This use case focuses on Karoline entering business expenses into the Quickbooks accounting system, allowing her to track them accurately.

2. Flow of Events

2.1 Basic Flow

1. Karoline clicks on the “Transactions” link
2. Karoline clicks on the “Record Transaction” button
3. Karoline enters Amount
4. Karoline enters Date
5. Karoline enters Type
6. Karoline enters Description
7. Karoline enters scanned receipt (optional)
8. Karoline clicks the “Submit” button
9. Quickbooks Database records transaction

3. Pre-conditions

3.1 Karoline has made a transaction that needs to be recorded

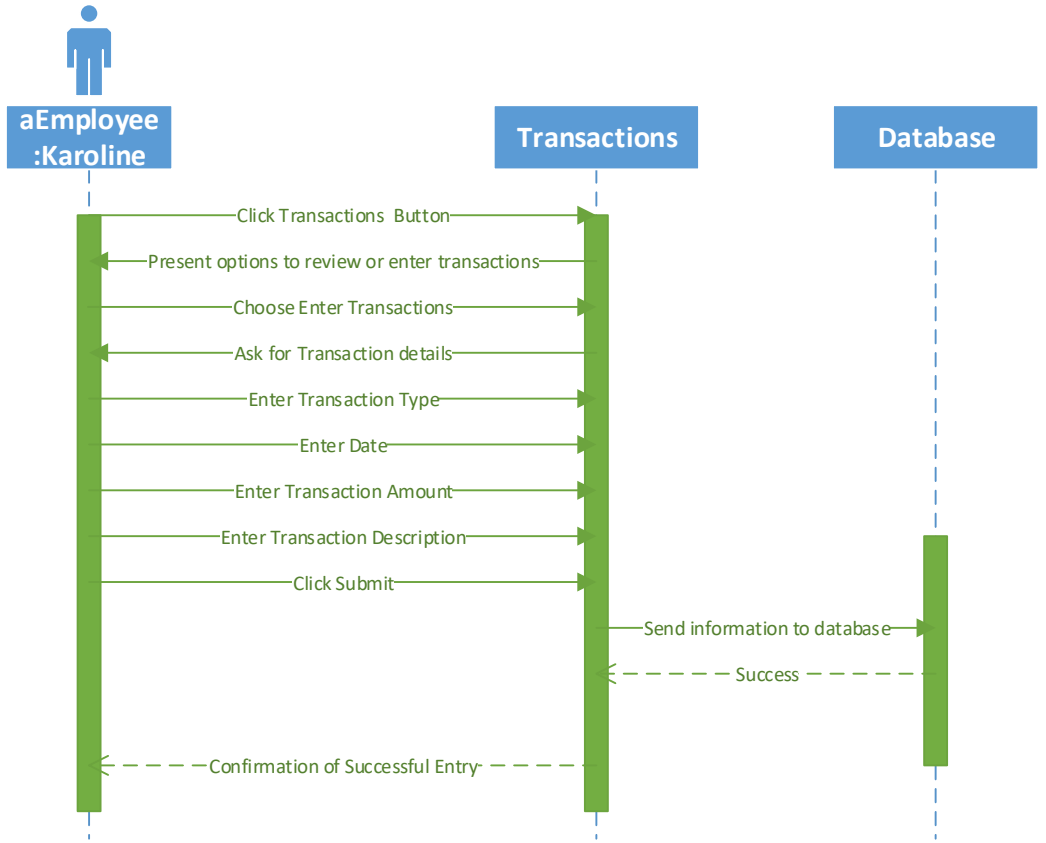
3.2 Karoline has logged into the system

4. Post-conditions

4.1 Financial reports are capable of being generated

Sequence Diagram: Enter Transactions

1. Karoline clicks on the “Transactions” link
2. Karoline clicks on the “Record Transaction” button
3. Karoline enters Amount
4. Karoline enters Date
5. Karoline enters Type
6. Karoline enters Description
7. Karoline enters scanned receipt (optional)
8. Karoline clicks the “Submit” button
9. Quickbooks Database records transaction



Use Case Specification: Forgot User Login

1. User Forgot Login Info

1.1 Brief Description

Use case for resolving issue when User cannot remember user name and/or password

2. Flow of Events

2.1 Basic Flow

User clicks on 'forgot user name' button

User types registered email address in email address text box

User click 'submit' button

System sends user name to registered email

2.2 Alternative Flows

2.2.1 Forgotten Password

User selects 'forgot password' button

User types registered email address in email address text box

User click 'submit' button

System sends temporary password to registered email

2.2.2 User doesn't have access to registered email

User clicks 'sign up for account' button

3. Special Requirements

No special requirements

4. Pre-conditions

No pre-conditions

5. Post-conditions

5.1 Temporary password

Temporary password is updated by user upon logging into the system

6. Extension Points

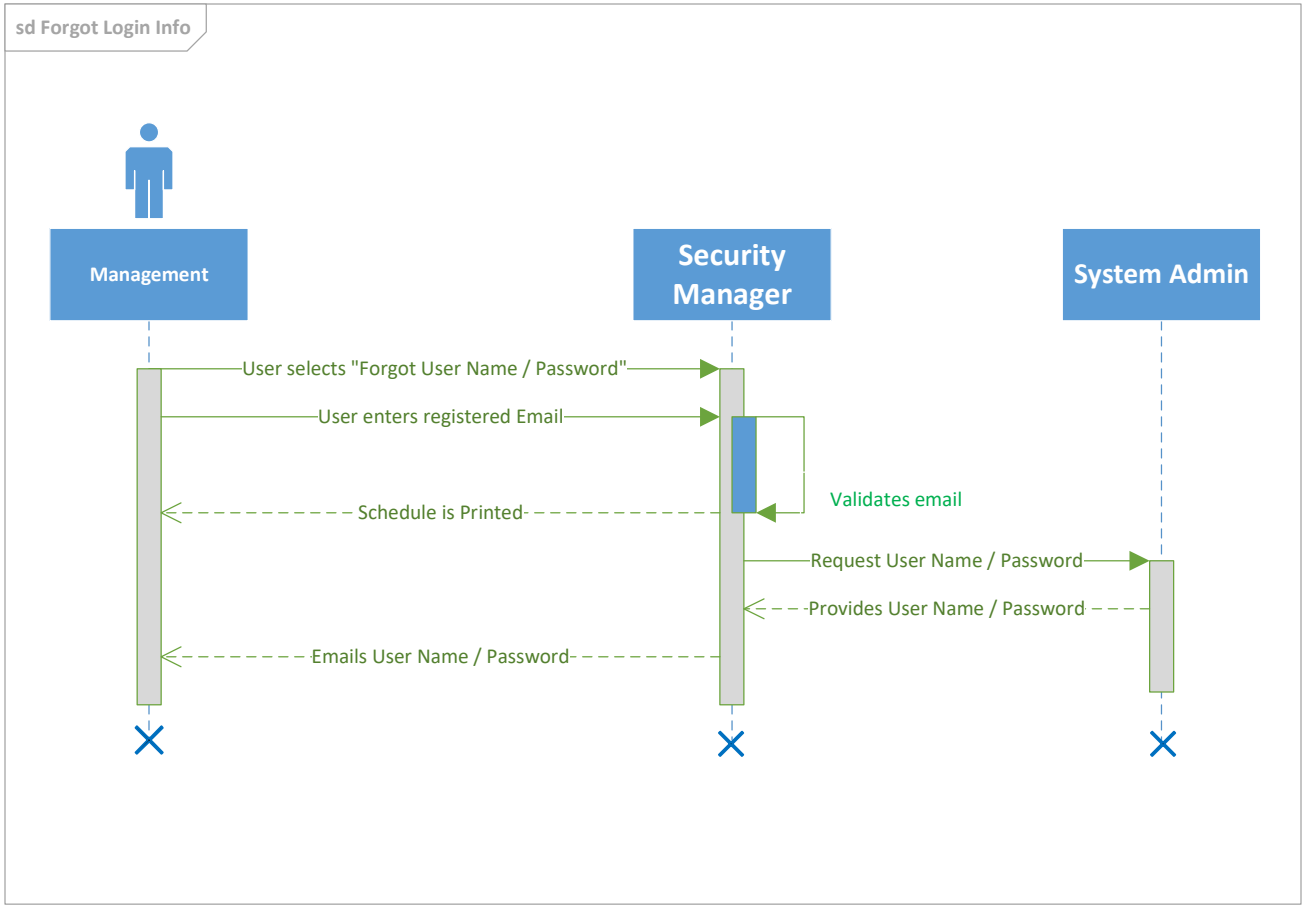
No Extension Points

SEQUENCE DIAGRAM: Forgot Login Info

MAIN FLOW

Normal Flow of Events:

1. User clicks on ‘forgot user name’ button
2. User types registered email address in email address text box
3. User click ‘submit’ button
4. System sends user name to registered email



Use Case Specification: Generate Backup Data

1. Use-Case Name

1.1 Brief Description

Backups should be performed regularly without needing to be initiated by Sweet Karoline's Cakes. The backups will be the responsibility of the company that hosts the database.

2. Flow of Events

2.1 Basic Flow

The Database Administrator of the cloud hosting chosen for the database will initiate all backups. The backups will be regular and will ensure fulfillment of the terms expressed in the contract. The DBA and the cloud hosting company will be responsible to backup the data without Karoline, or any other agent on behalf of Sweet Karoline's Cakes, initiating the backup.

2.2 Alternative Flows

2.2.1 None

3. Special Requirements

3.1 Digital Security of Personally Identifiable Information

Extra steps for digital security should be taken to protect the PII. Encryption of data is suggested.

4. Pre-conditions

4.1 The Contract Needs to be Agreed Upon, and Signed

The contract must specify the duties, responsibilities, and must be agreed upon by Karoline and the cloud hosing company.

5. Post-conditions

5.1 All Site and Company Data is Backed up to an Indexed Image

The data is recoverable by choosing the created backup based on the date and time that the backup was created.

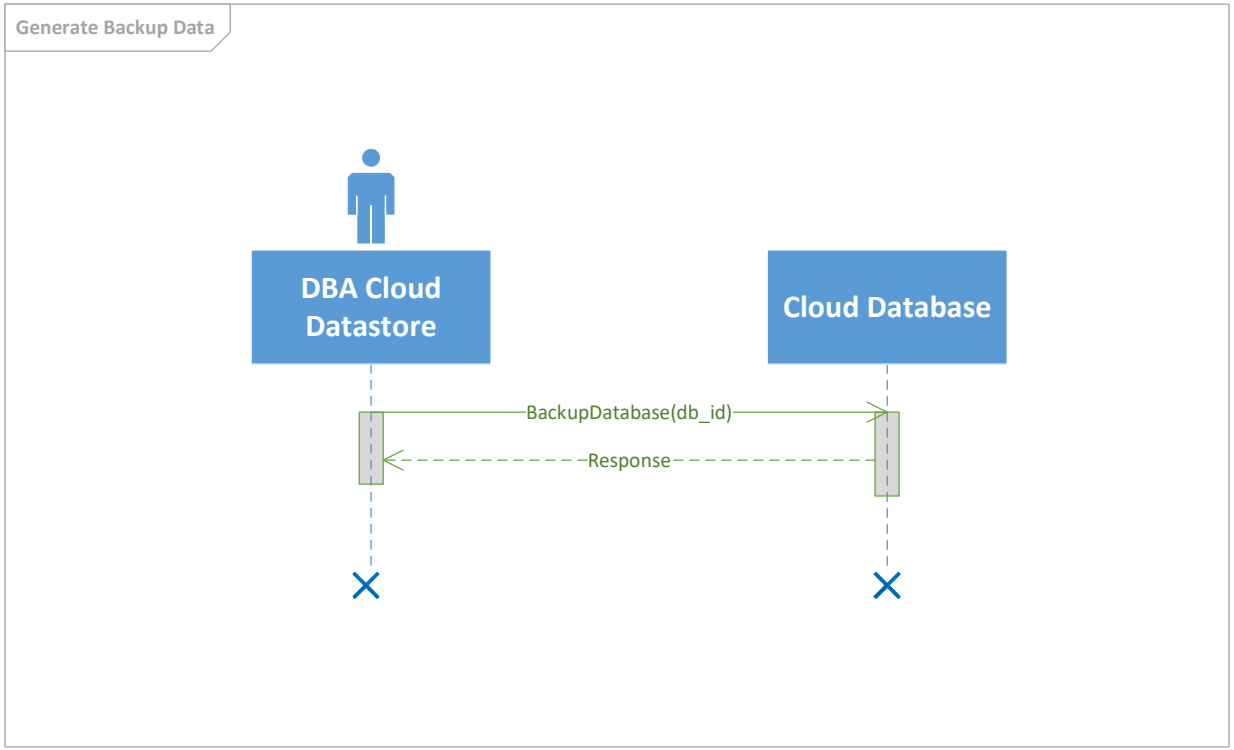
6. Extension Points

6.1 None

SEQUENCE DIAGRAM: Generate Backup Data

MAIN FLOW

- The Database Administrator of the cloud hosting chosen for the database will initiate all backups.
- The backups will be regular and will ensure fulfillment of the terms expressed in the contract.
- The DBA and the cloud hosting company will be responsible to backup the data without Karoline, or any other agent on behalf of Sweet Karoline's Cakes, initiating the backup.



Use Case Specification: Generate Financial Reports

1. Use-Case Name

1.1 Brief Description

This use case describes the generation of financial reports, including financial statements, by Karoline after a full accounting of all financial transactions has been made.

2. Flow of Events

2.1 Basic Flow

Karoline navigates to the "Reports" section of Quickbooks
Karoline chooses the type of report to run
Karoline chooses the date range she wants to use
Karoline selects a filter on transaction amount, if necessary
Karoline selects a filter on transaction type, if necessary
Karoline selects a filter on transaction description, if necessary
Karoline clicks the "Generate Report" button

3. Pre-conditions

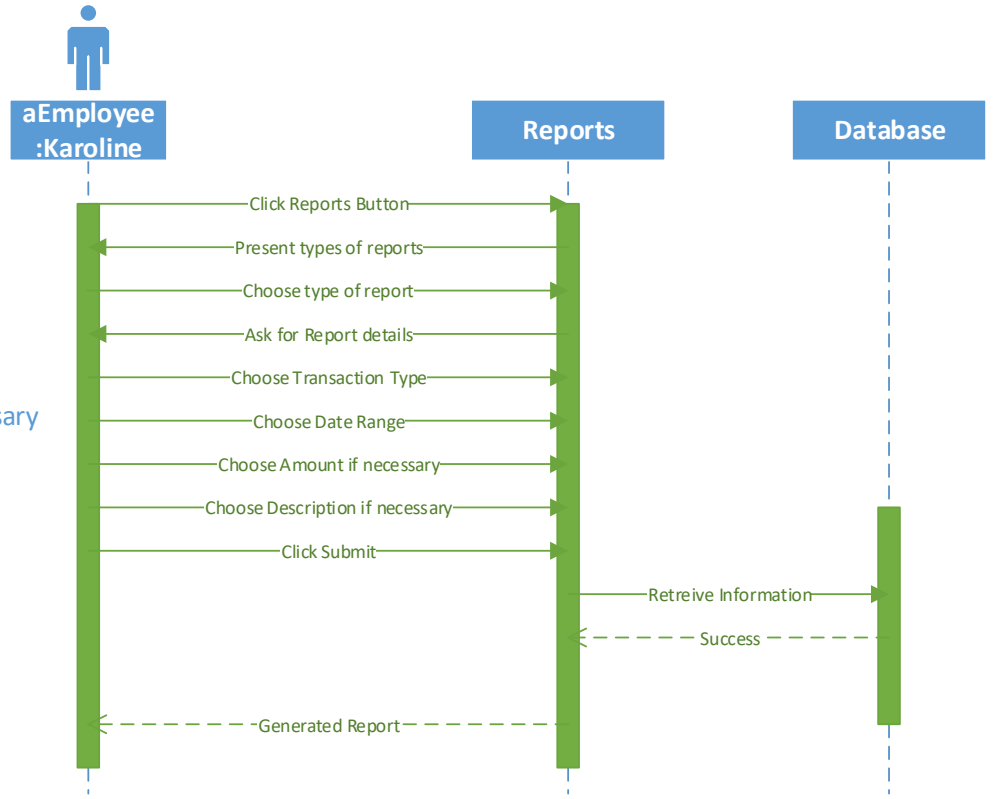
3.1 All revenues and expenses for the report period have been entered

4. Post-conditions

4.1 Financial reports are ready to print

Sequence Diagram: Generate Reports

1. Karoline navigates to the “Reports” section of Quickbooks
2. Karoline chooses the type of report to run
3. Karoline chooses the date range she wants to use
4. Karoline selects a filter on transaction amount, if necessary
5. Karoline selects a filter on transaction type, if necessary
6. Karoline selects a filter on transaction description, if necessary
7. Karoline clicks the “Generate Report” button



Use Case Specification: Generate List of Ingredients

1. Generate List of Ingredients

1.1 Brief Description

This use case is for generating a list of ingredients for Sweet Karoline's Cakes inventory.

2. Flow of Events

2.1 Basic Flow

User creates order

Karoline approves order

An automated list of ingredients is generated based on what the user orders.

Use case ends

3. Pre-conditions

3.1 User has account, user makes order.

The user must have an account on the Sweet Karoline's Cakes website and must create an order.

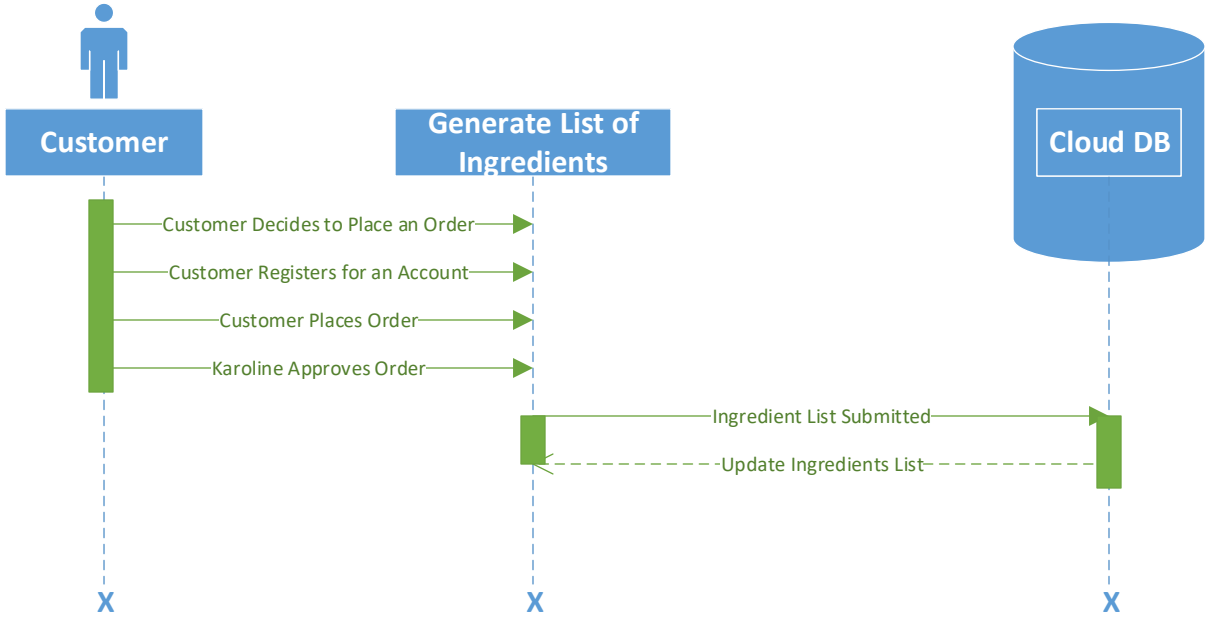
4. Post-conditions

4.1 Ingredient List is generated

The list of ingredients is compiled and ready for further action.

Generate List of Ingredients

- The Customer decides to place an order.
- The Customer registers for an Account.
- The Customer places the order.
- Karoline Approves the order.
- The Ingredient List is submitted to the Database.
- The Database updates the Ingredient List.



Use Case Specification: Jump to social media site

1. Jump to social media site

1.1 Brief Description

A customer will be able to click on a link and go from sweet Karoline's cakes website to her social media site.

2. Flow of Events

2.1 Basic Flow

Actors: Customer

User will click on the Facebook thumbnail.

Or user will click on the Instagram thumbnail.

A new tab opens.

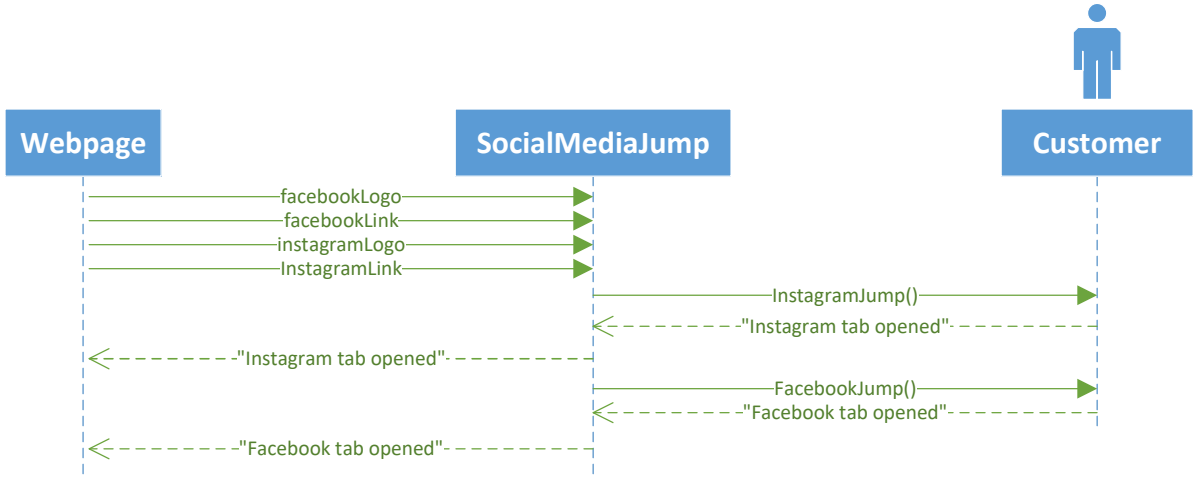
The new tab is populated with the respective social media page.

3. Post-conditions

3.1 User taken to social media site

After the use case, a new tab is open to Karoline's social media page.

Main flow of events:
1. Webpage show thumbnail for social media site.
2. Customer clicks on related thumbnail.
3. Customer is taken to social media site.



Use Case Specification: Print Financial Statements

1. Use-Case Name

1.1 Brief Description

This use case describes the process of printing the financial statements once they have been completed.

2. Flow of Events

2.1 Basic Flow

1. Karoline selects Print Report
2. Karoline selects printer from menu
3. Computer checks to see if printer is attached
4. Karoline clicks the "Print" button

3. Special Requirements

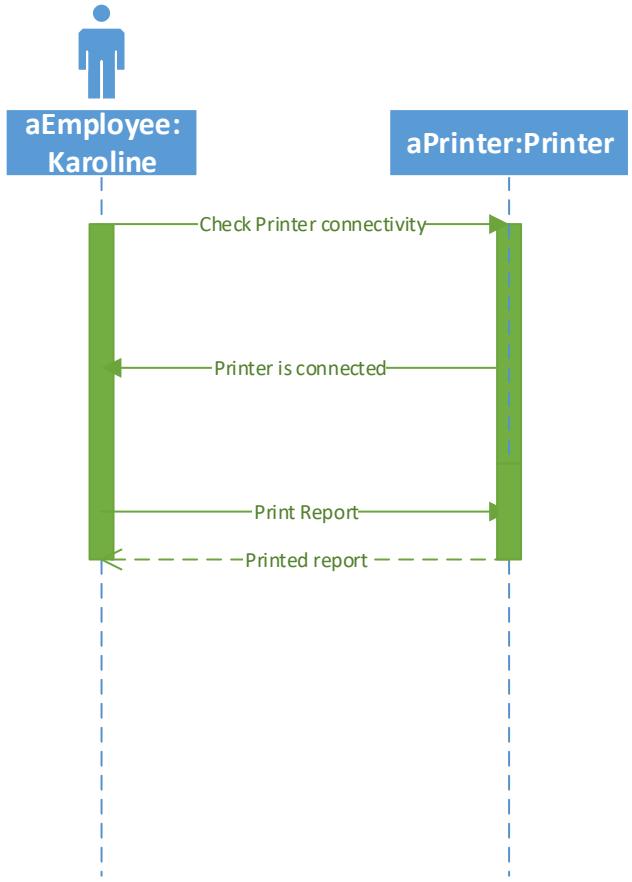
3.1 Must have a printer in order to print reports

4. Pre-conditions

4.1 Financial Statements/Reports have been generated

Sequence Diagram: Print Reports

1. Karoline selects Print Report
2. Karoline selects printer from menu
3. Computer checks to see if printer is attached
4. Karoline clicks the “Print” button



Use Case Specification: Print Marketing Reports

1. Print Marketing Reports

1.1 Brief Description

Karoline may print marketing reports based on reports that are automatically generated or reports that are manually generated. Report(s) run dates may be predetermined or printed when Karoline decides.

2. Flow of Events

2.1 Basic Flow

Actor(s): Karoline

Karoline sets frequency of automatically generated reports and they will print when she confirms the reports are complete.

2.2 Alternative Flows

2.2.1 Manually Printed Reports

Actor(s): Karoline

Karoline can select both completed and uncompleted marketing reports to print at her discretion. Reports will be compiled and printed at this time.

3. Special Requirements

No Special Requirements

4. Pre-conditions

4.1 Marketing Reports have been finalized

Automatically generated reports must be finalized before printing can occur.

5. Post-conditions

No Post Conditions

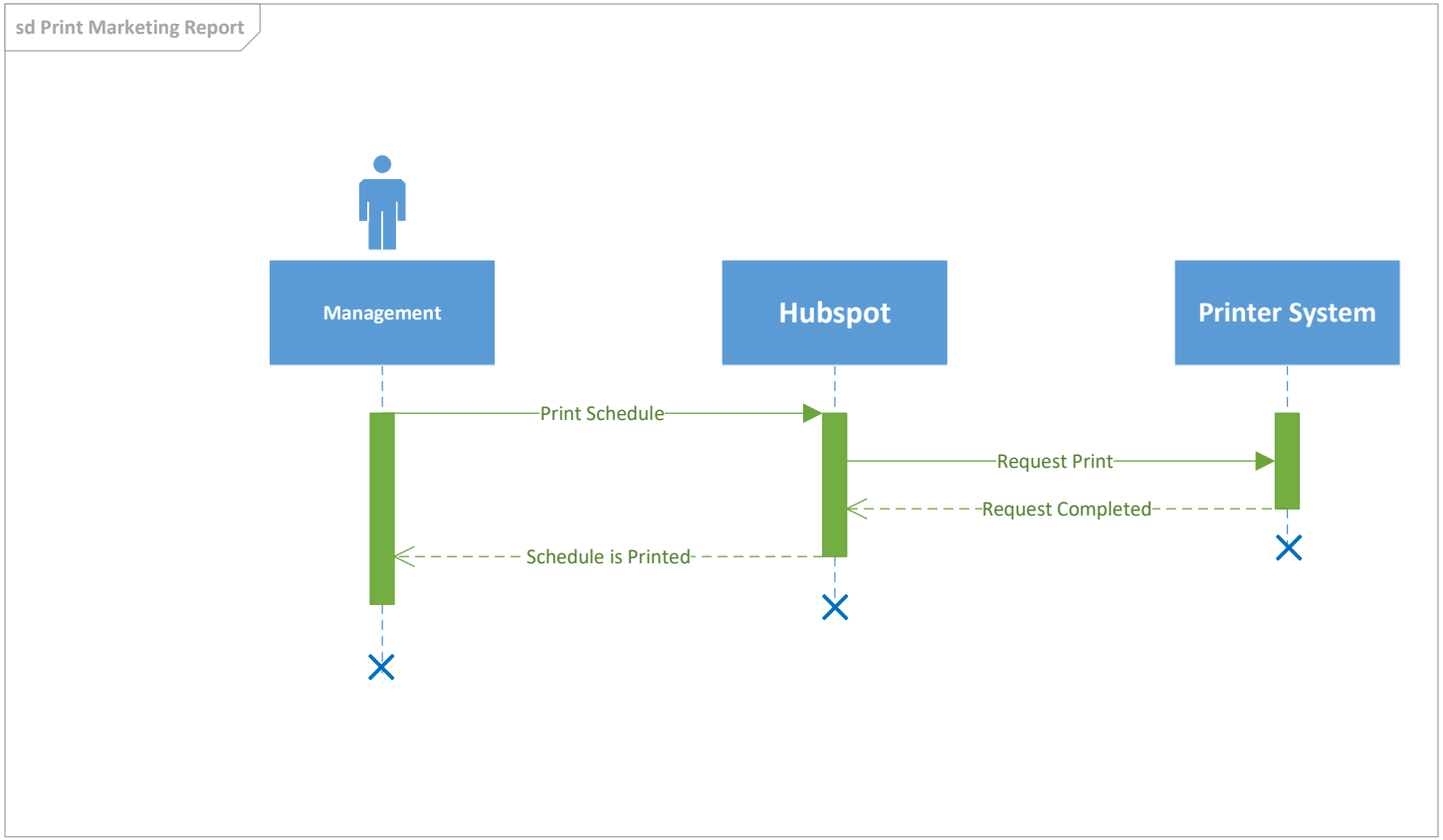
6. Extension Points

No Extension Points

SEQUENCE DIAGRAM: Print Marketing Report

MAINFLOW

1. User Selects Marketing Reports
2. User Prints Marketing Reports



Use Case Specification: Print Weekly Schedule

1. Print Weekly Schedule

1.1 Brief Description

This use case describes how the user (Karoline) would print out the weekly schedule of her cake appointments.

2. Flow of Events

2.1 Basic Flow

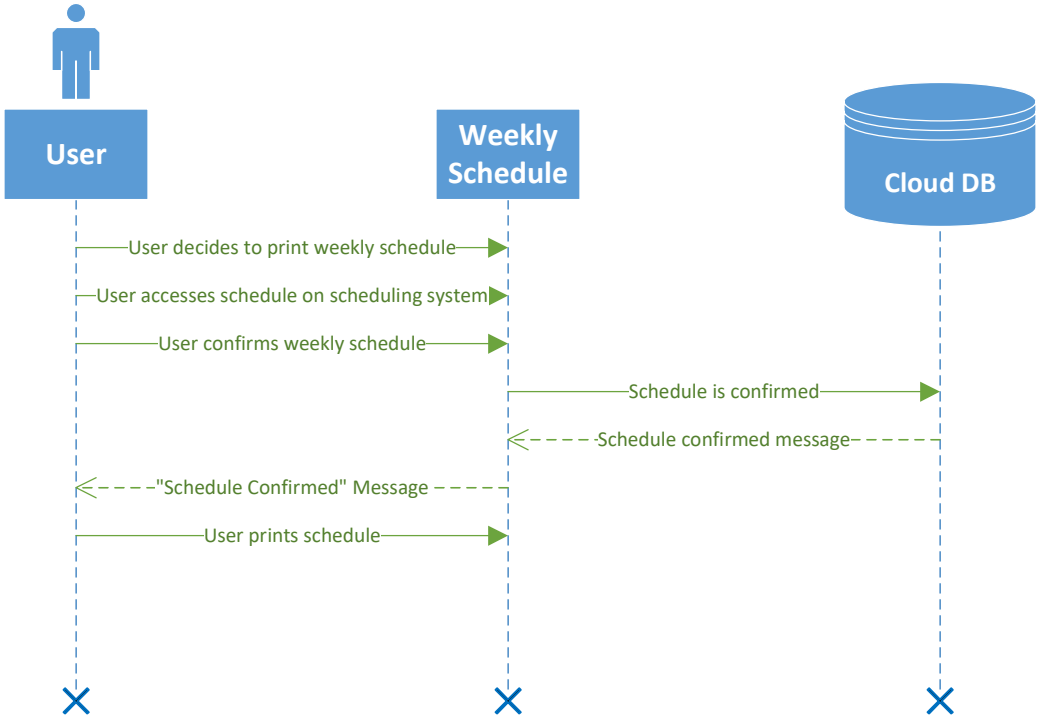
1. User (Karoline) decides to print out the weekly schedule.
2. User pulls up schedule on scheduling system.
3. User confirms weekly schedule.
4. User prints weekly schedule.

3. Pre-conditions

3.1 Appointments have been requested then confirmed in the scheduling system.

Print Weekly Schedule

1. User (Karoline) decides to print out the weekly schedule.
2. User pulls up schedule on scheduling system.
3. User confirms weekly schedule.
4. User prints weekly schedule.



Use Case Specification: Produce Marketing Reports

1. Produce Marketing Reports

1.1 Brief Description

Karoline will utilize the CRM program to record marketing information and generate reports based on this information.

2. Flow of Events

2.1 Basic Flow

Actors: Karoline, CRM program

When Karoline wants to view, print, or present marketing information, such as Campaign performance or Conversion metrics, she will use the CRM software to generate reports either automatically at predetermined times (i.e. monthly, quarterly, annually, etc.).

2.2 Alternative Flows

2.2.1 Karoline generates a manual report

Actors: Karoline, CRM

Karoline may use CRM program to select some or all sections of the Marketing report.

These reports can be generated, printed, and presented and any time interval.

3. Special Requirements

There are no Special Requirements

4. Pre-conditions

4.1 Market data must be captured and stored by the CRM program

For the marketing data to be useful, all data must be tracked through the CRM program.

5. Post-conditions

5.1 Report is printed

Report(s) is printed out at a predetermined time or at the discretion of Karoline.

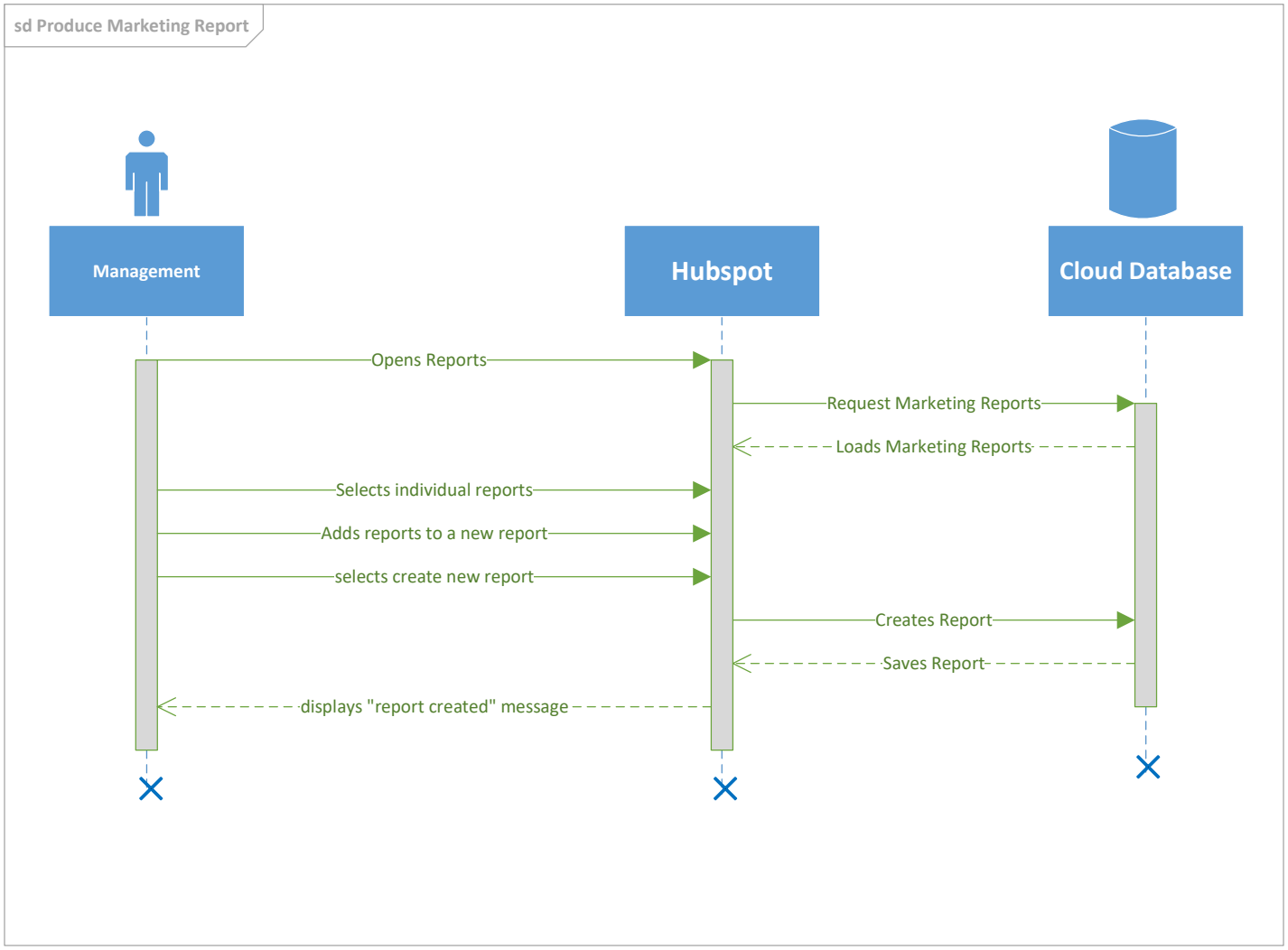
6. Extension Points

6.1 No Extension Points

SEQUENCE DIAGRAM: Produce Marketing Report

MAIN FLOW

1. Management selects ‘Open Reports’ button on CRM dashboard
2. Selects report(s) from listbox
3. Management adds report(s) from listbox to Report listbox
4. Creates New Report
5. System saves reports



Use Case Specification: Receive order or quote request

1. Receive order or quote request

1.1 Brief Description

When a customer has completed a quote or an order, Karoline will be notified and given all the information.

2. Flow of Events

2.1 Basic Flow

Actors: Management

User receives an email.

Email contains Customers first name.

Email contains Customers last name.

Email contains Customers telephone number.

Email contains Customers email address.

Email contains Customers product.

Email contains Customers quantity.

Email contains Customers completion date.

Email contains pickup or delivery.

If delivery email contains delivery street address.

If delivery email contains delivery city.

If delivery email contains delivery state.

If delivery email contains delivery zip code.

2.2 Alternative Flow

2.2.1 User completed quote request

Actors: Management

User receives an email.

Email contains Customers first name.

Email contains Customers last name.

Email contains Customers telephone number.

Email contains Customers email address.

Email contains Customers serving size.

Email contains Customers tier count.

Email contains Customers budget.

Email contains Customers cake flavors.

Email contains Customers cake filling.

Email contains Customers frosting flavors

Email contains Customers any addition comments.

If customer uploaded image, Email contains image.

Email contains Customers completion date.

Email contains pickup or delivery.

If delivery email contains delivery street address.

If delivery email contains delivery city.

If delivery email contains delivery state.

If delivery email contains delivery zip code.

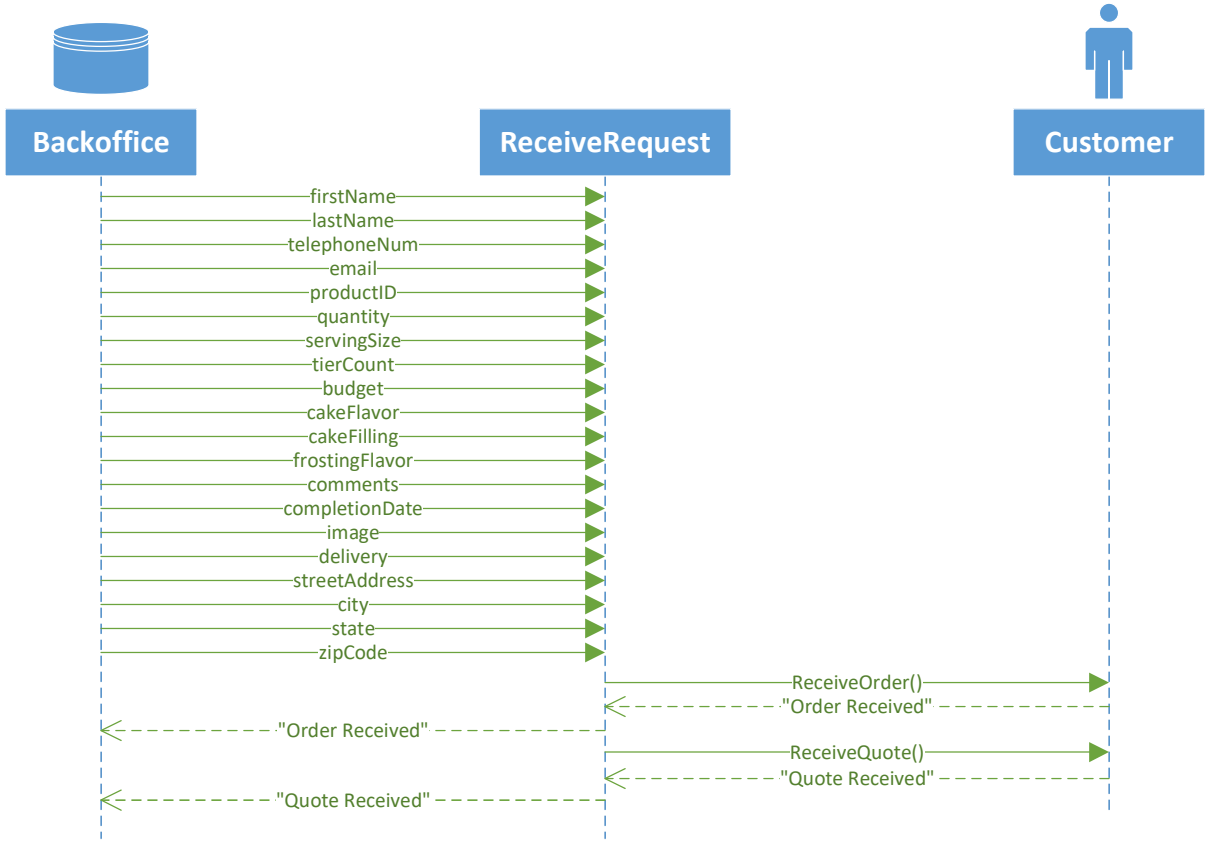
3. Pre-conditions

3.1 Customer has completed request

Before the request can be received by Karoline, it must first be submitted by the customer.

Main flow of events:

1. System populates order information into form.
2. System sends form to administrator.
3. Administrator views form.



Use Case Specification: Recover Backup Data

1. Use-Case Name

1.1 Brief Description

This use case is for restoring data from a backup image that is saved to a cloud.

2. Flow of Events

2.1 Basic Flow

Sweet Karoline's Cakes (SKC) contacts the point of contact for the database hosting service. SKC provides the Database ID. SKC requests that a backup be performed at a specific time. SKC requests that the data be restored to a backup image. The cloud database DBA performs the restoration of the data to the chosen image.

2.2 Alternative Flows

2.2.1 No alternatives

3. Special Requirements

3.1 No Special Requirements

4. Pre-conditions

4.1 The Selected Backup Data Source Exists

The selected source for the data backup must already exist. The database hosting company must make regular backups to ensure the ability to back up the data.

5. Post-conditions

5.1 The Selected Data is Restored from The Selected Backup Source

The data that was selected for restoration is used for recovery. Data will reflect the data at the time the backup image was made.

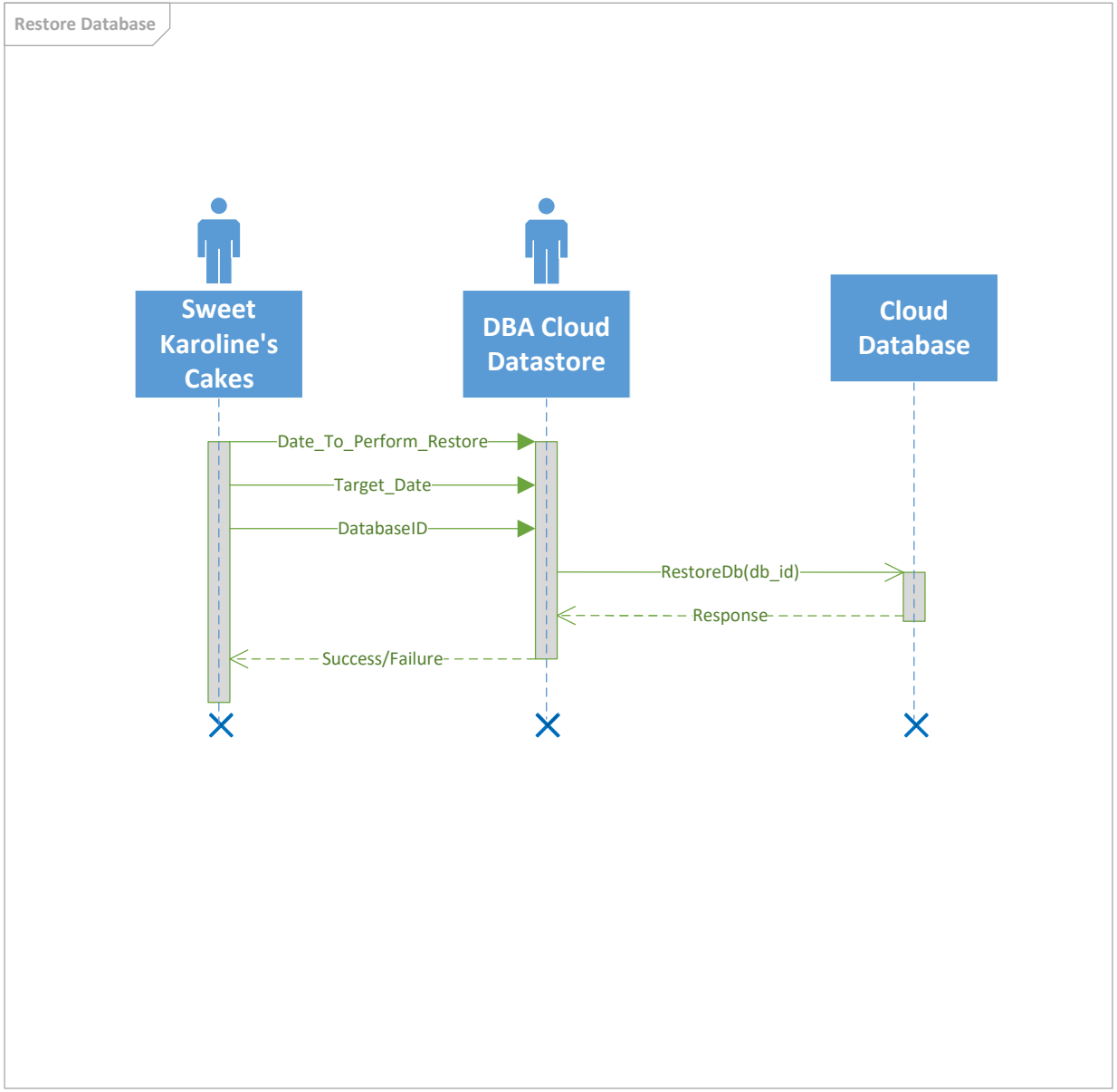
6. Extension Points

6.1 None

SEQUENCE DIAGRAM: Recover Backup Data

MAIN FLOW

- Sweet Karoline's Cakes (SKC) contacts the point of contact for the database hosting service.
- SKC provides the Database ID.
- SKC requests that a backup be performed at a specific time.
- SKC requests that the data be restored to a backup image.
- The cloud database DBA performs the restoration of the data to the chosen image.



Use Case Specification: Schedule Completion Date

1. Schedule Completion Date

1.1 Brief Description

This use case describes how a customer would go about scheduling an appointment on the Sweet Karoline's Cakes website.

2. Flow of Events

2.1 Basic Flow

1. Customer visits Sweet Karoline's Cakes website
2. Customer decides to order a cake/cupcake.
3. Customer completes steps in order request form, subsequently arriving at the scheduling section of the form.
4. Scheduling system provides the customer with a calendar showing open dates and times that are available for a cake to be completed and delivered on.
5. Customer chooses an appointment that works best for them from the available dates shown
6. Customer submits their date/time request.
7. Scheduling system alerts Karoline Gardner of a new appointment being made.

2.2 Alternative Flows

2.2.1 Customer Chooses Carryout

1. Scheduling system provides alternative dates and times for which their cake is available to be completed and ready for pickup by.
2. Customer chooses date and time from those available for carryout
3. Customer submits scheduling request.

3. Pre-conditions

3.1 Customer has completed a quote request form.

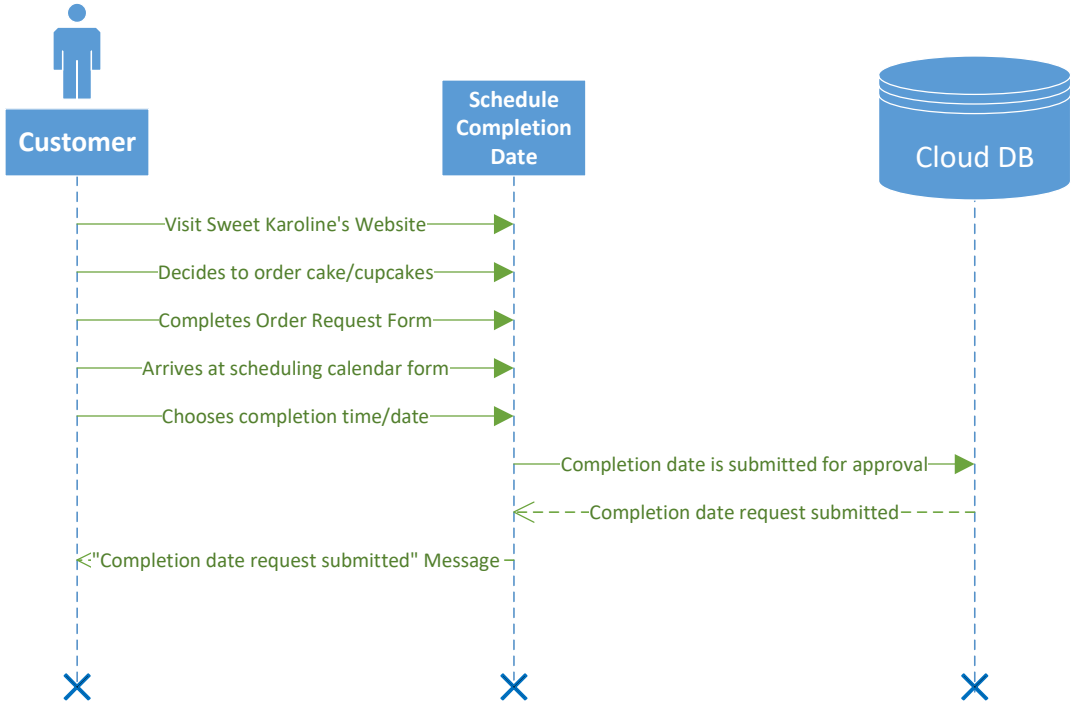
4. Post-conditions

4.1 Karoline either confirms or denies the appointment date chosen.

4.2 A receipt is sent back if confirmed; customer is told to choose another date/time if denied.

Schedule Completion Date

1. Customer visits Sweet Karoline’s Cakes website
2. Customer decides to order a cake/ cupcake.
3. Customer completes steps in order request form, subsequently arriving at the scheduling section of the form.
4. Scheduling system provides the customer with a calendar showing open dates and times that are available for a cake to be completed and delivered on.
5. Customer chooses an appointment that works best for them from the available dates shown
6. Customer submits their date/time request.
7. Scheduling system alerts Karoline Gardner of a new appointment being made.



Use Case Specification: Submit Online Payment

1. Use-Case Name

1.1 Brief Description

The purpose of this use case is to describe the process by which a customer enters payment information on the Sweet Karoline's website. In addition, it will describe the systems used and the pre-/post-conditions of the process, along with any other requirements that are necessary to ensure the payment process is conducted correctly and efficiently.

2. Flow of Events

2.1 Basic Flow

1. Customer Clicks Pay button
2. Customer chooses method of payment; card assumed for main flow
3. Customer enters name
4. Customer enters card number
5. Customer enters expiration date
6. Customer enters security code
7. Customer enters billing address
8. Customer enters billing city
9. Customer enters billing state
10. Customer enters billing zip
11. Customer verifies information
12. Customer clicks submit button
13. Payment is transferred to Paypal for processing
14. Paypal sends confirmation of payment to customer

2.2 Alternative Flows

2.2.1 Pay by method other than card

Should the customer choose to pay by a method other than a credit/debit card, they will instead provide payment in the appropriate form (cash/check) upon delivery of the purchased items.

3. Pre-conditions

In order to be take to the payment screen to submit a paymnt, a customer must have already signed in to the website (if applicable) and created an order.

3.1 Customer Signs In to Website

3.2 Customer Creates Order

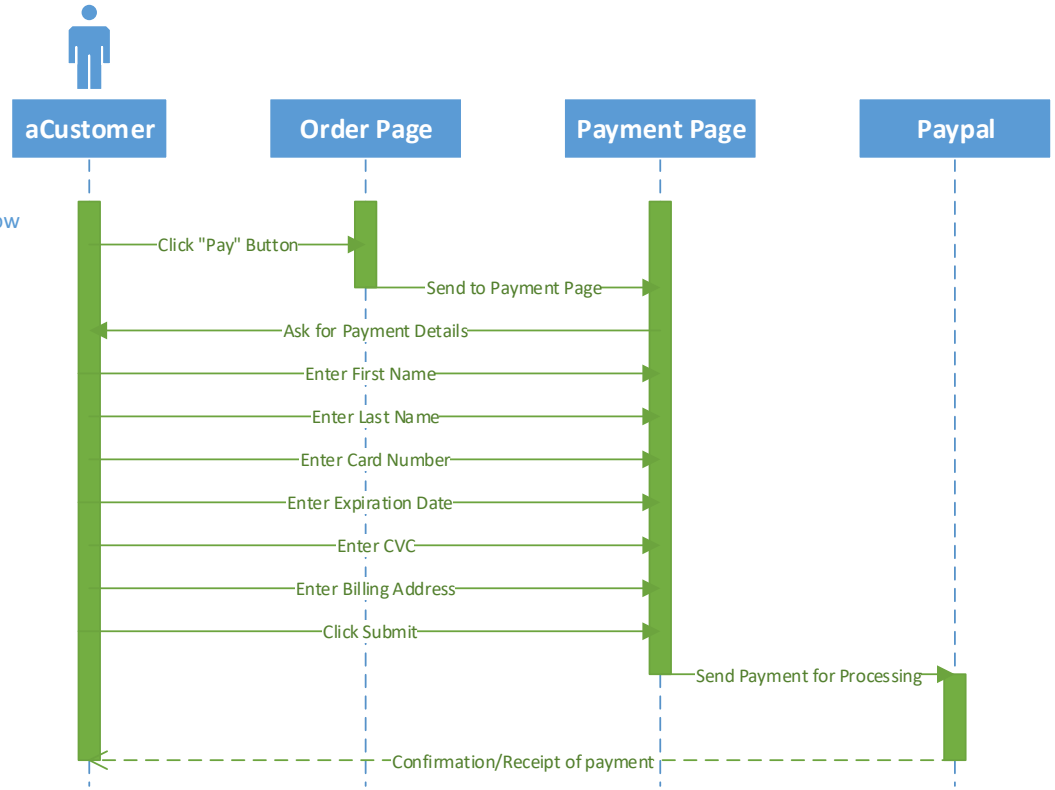
4. Post-conditions

Following the submission of payment, the system will move to processing the payment, which will be handled by Paypal. Once this payment has been processed, Karoline will have the necessary information to begin generating revenue reports and tracking her sales.

4.1 Karoline Tracks Revenues Gained

Sequence Diagram: Submit Payment

1. Customer Clicks Pay button
2. Customer chooses method of payment; card assumed for main flow
3. Customer enters name
4. Customer enters card number
5. Customer enters expiration date
6. Customer enters security code
7. Customer enters billing address
8. Customer enters billing city
9. Customer enters billing state
10. Customer enters billing zip
11. Customer verifies information
12. Customer clicks submit button
13. Payment is transferred to Paypal for processing
14. Paypal sends confirmation of payment to customer



Use Case Specification: Update Inventory

1. Update Inventory

1.1 Brief Description

This use case shows how the user (Karoline) will manually update inventory.

2. Flow of Events

2.1 Basic Flow

Karoline purchases inventory

Karoline enters new ingredients into inventory

Karoline saves inventory entry

2.2 Alternative Flows

2.2.1 Karoline Removes Ingredients from Inventory

Karoline removes extra used ingredients from inventory

Karoline saves inventory edit

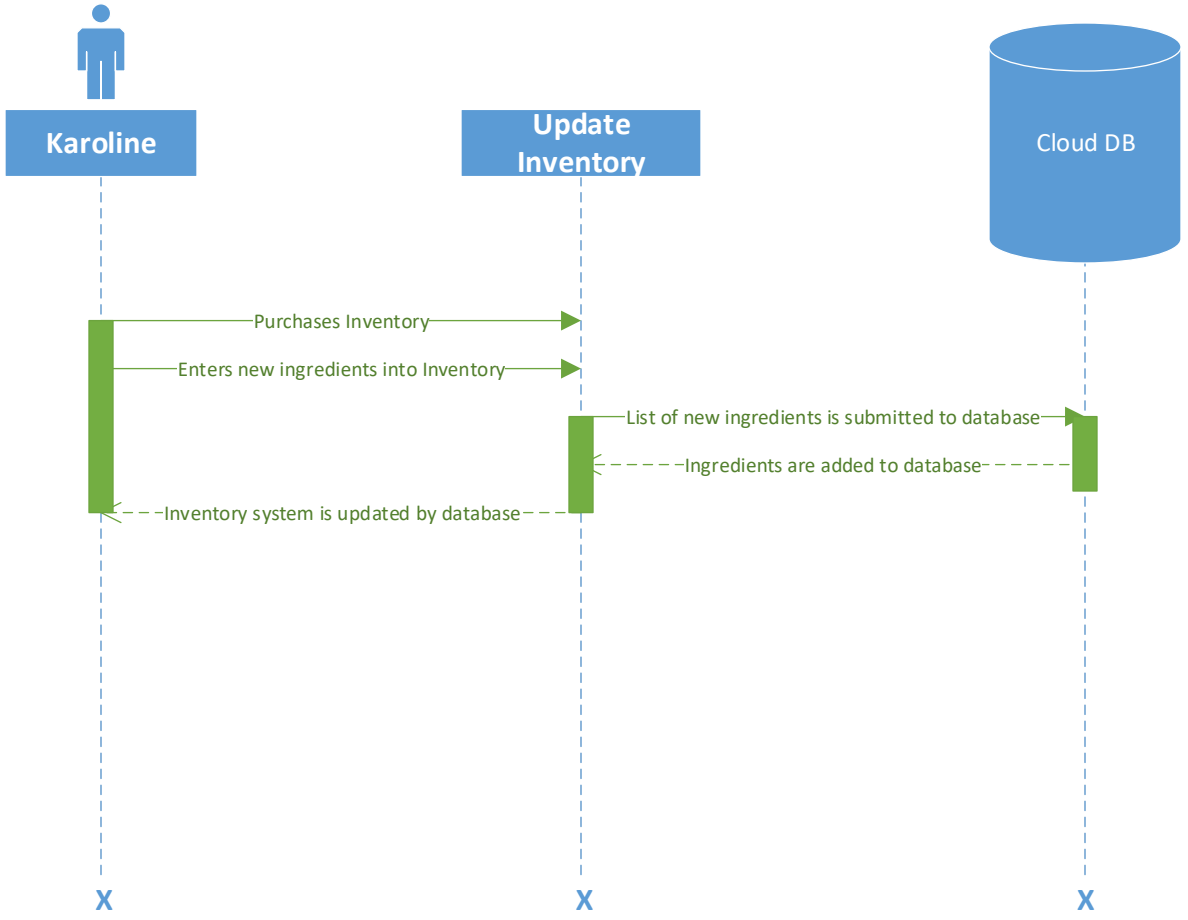
3. Pre-conditions

3.1 Inventory must exist

Inventory must already exist to update.

Update Inventory

- Karoline purchases inventory.
- Karoline enters the new ingredients into inventory.
- The new list of ingredients is submitted to the database.
- The ingredients are added to the database.
- The inventory system is updated by the database.



Use Case Specification: View previous items

1. View previous items

1.1 Brief Description

A customer will be able to view pictures of previously made items on the website.

2. Flow of Events

2.1 Basic Flow

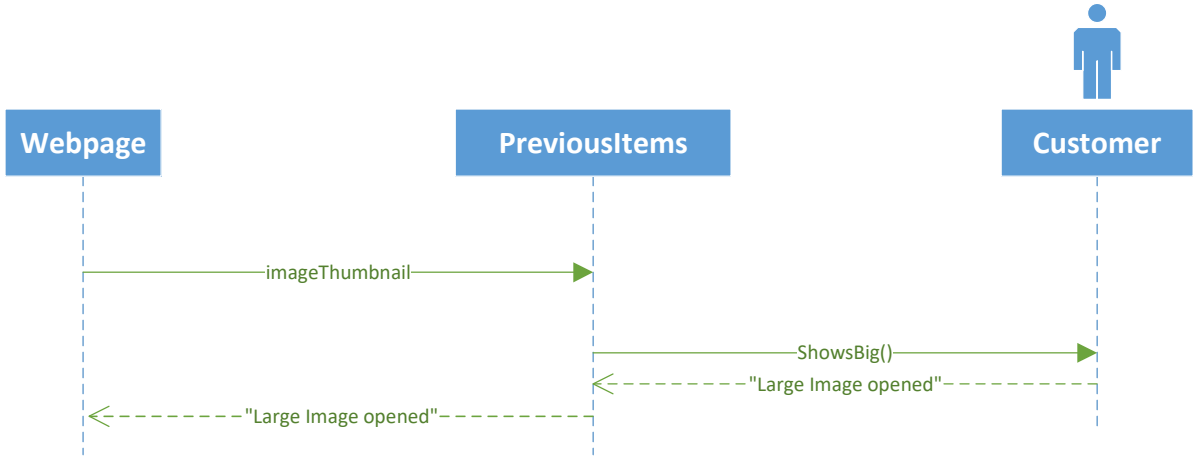
Actors: Customer

User will click on the gallery tab on the navigation bar.

User will see a webpage of multiple smaller images of products Karoline has previously created.

User clicks on the image to get a larger version of the image.

Main flow of events:
1. Webpage generate thumbnail of previous item.
2. Customer clicks on thumbnail.
3. Large image of thumbnail opens for view.



Use Case Specification: View Product Information

1. View Product Information

1.1 Brief Description

This use case will show how the user can view product information for the variety of products Sweet Karoline's Cakes offers.

2. Flow of Events

2.1 Basic Flow

User scrolls down Menu to view prices for:

- Servings
- Tiers
- Cupcakes
- Cookie Cakes
- Pies
- Cake Pops
- Cakes with fresh fruit

User scrolls down further to view options for:

- Cake Flavors
- Cake Fillings
- Frosting Flavors

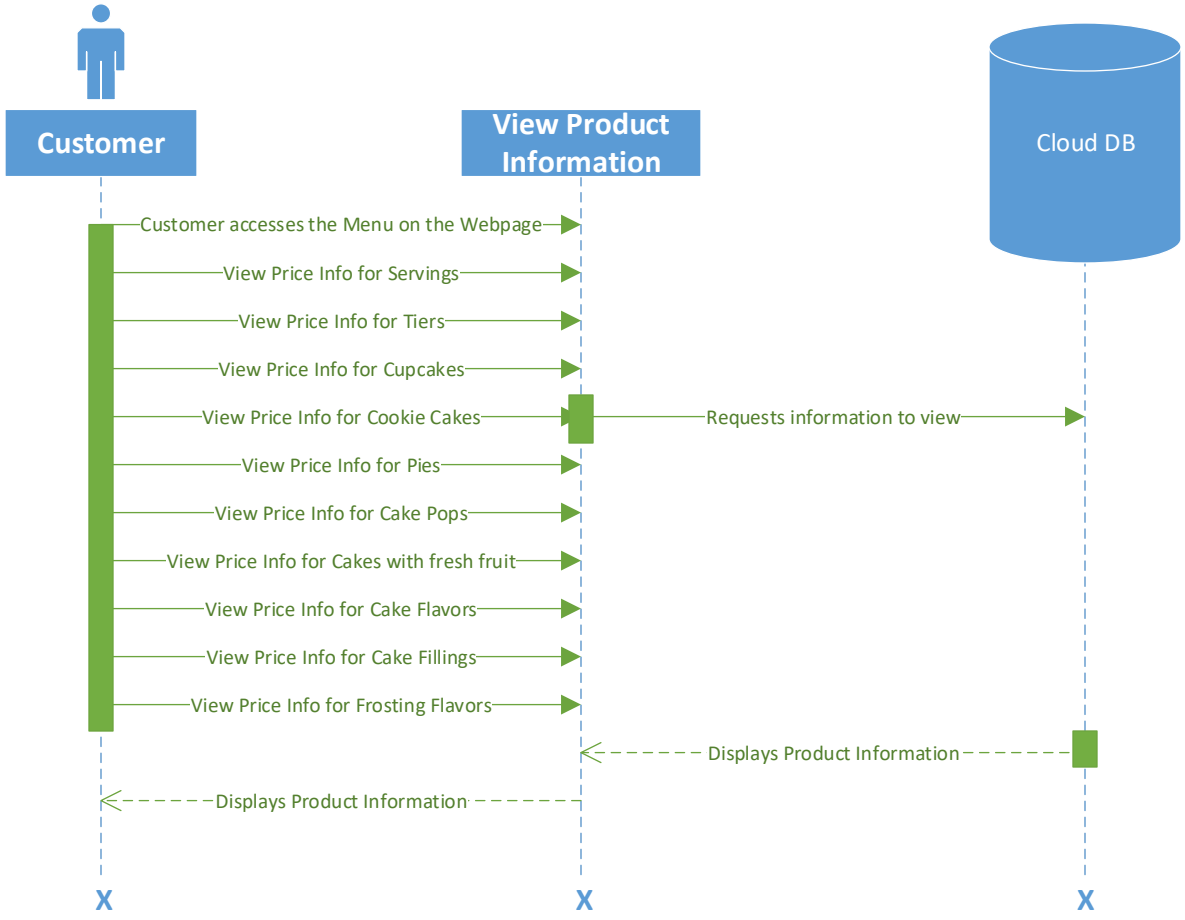
3. Pre-conditions

3.1 User has internet access

User must have a computer and access to the internet to access the Sweet Karoline's Cakes website.

View Product Information

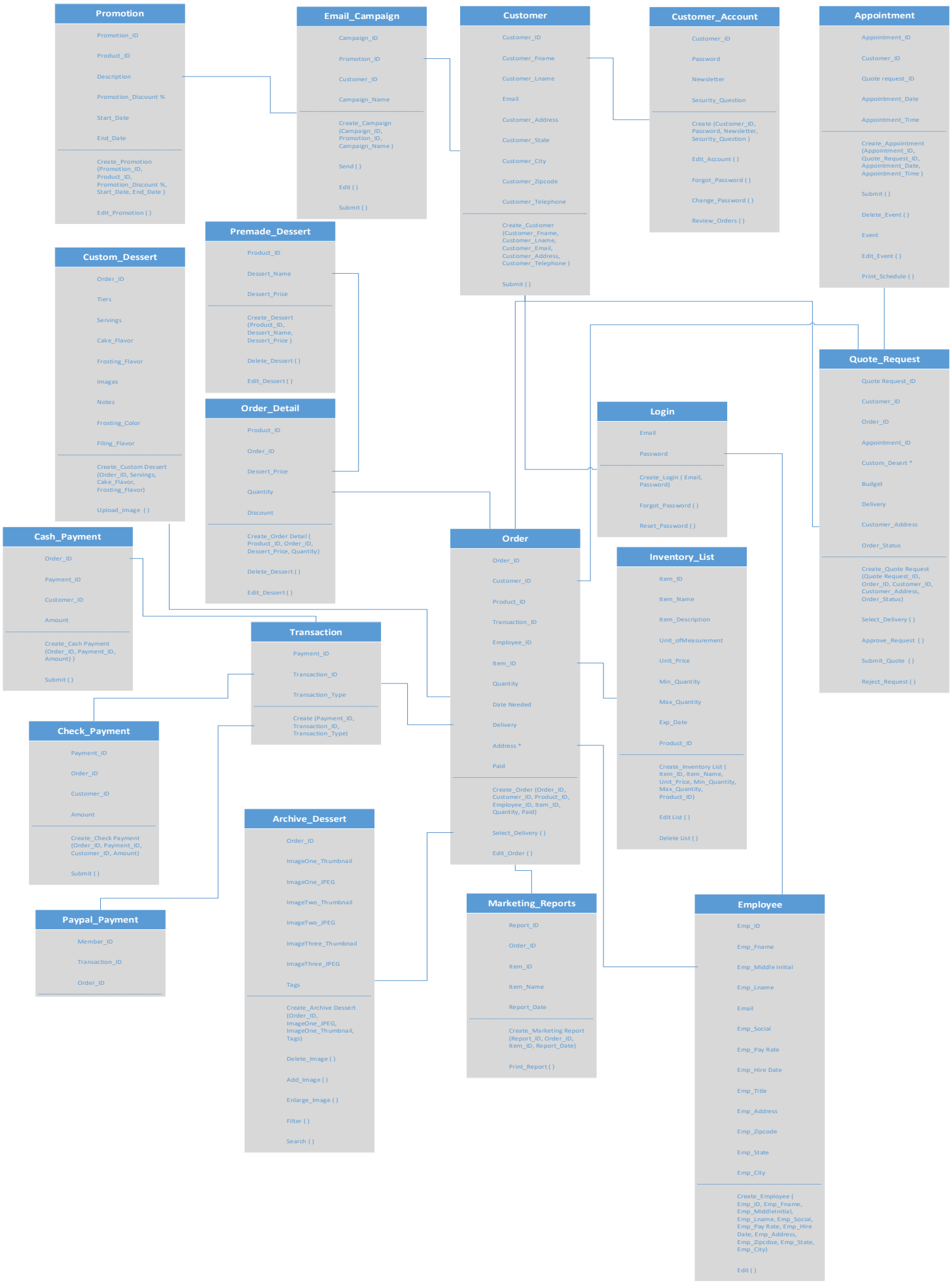
- Customer accesses the Menu on the webpage.
- They can view pricing information for: servings, tiers, cupcakes, cookie cakes, pies, cake pops, cakes with fresh fruit, cake flavors, cake fillings, and frosting flavors.
- The information is requested for viewing.
- The database displays the product information to the customer.



Class Diagram Narrative

A class diagram is a mapping of the classes, attributes, and methods that make up the system being designed. So, in layman's terms, it shows the general makeup of the system, the parts that make it work and the ways said parts communicate with each section of the system. For example, how address connects with customer account and how the user has the ability to change details of the address, which is then applied to the overarching customer account class.

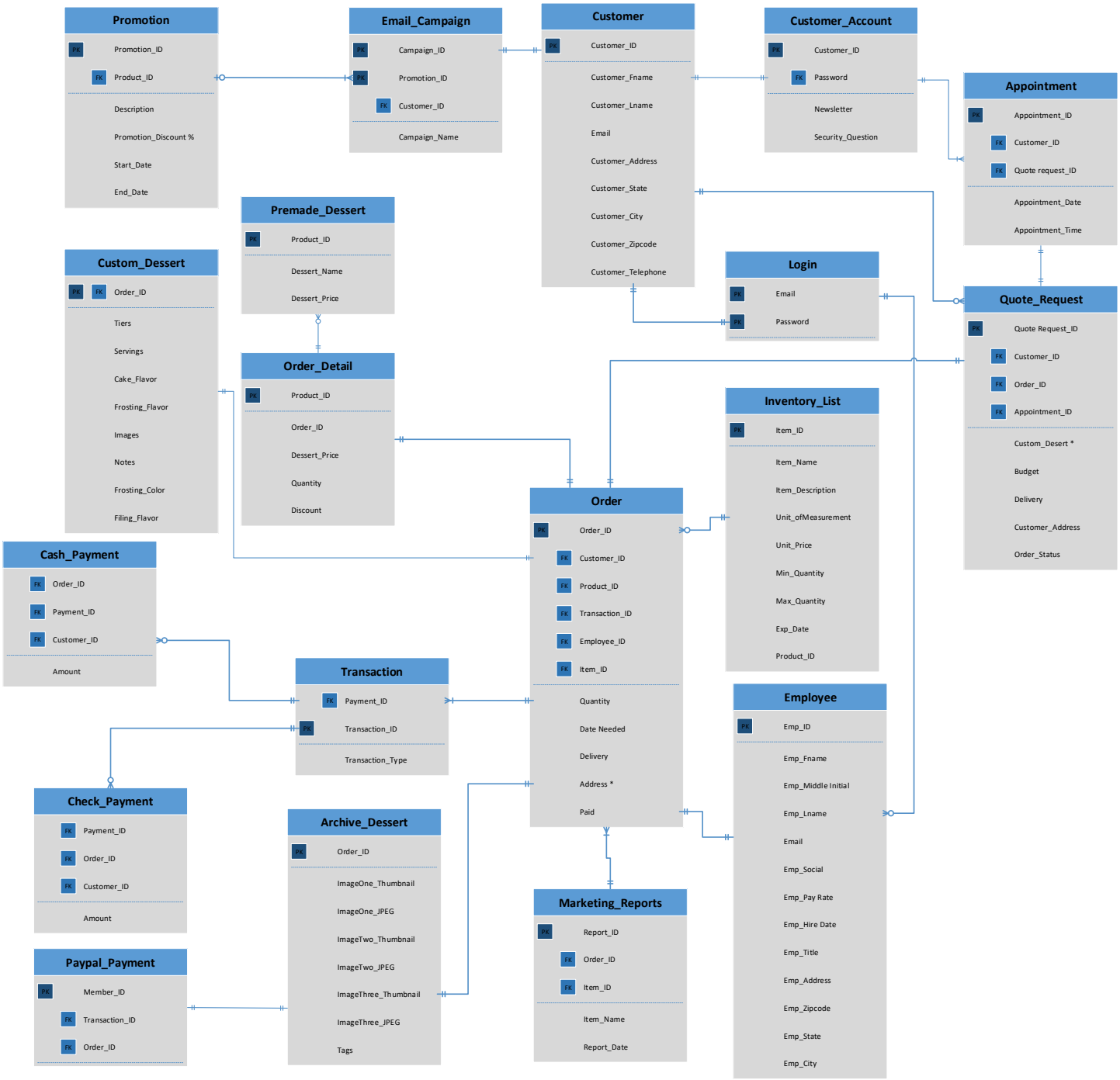
The class diagram was derived from prototype analysis, which is a method of development where the scope and aspects of a system are developed, changed, added, etc. as the iterative process of developing the prototype continues. Improvements and potential ideas are analyzed as the prototype takes form, and the class diagram is developed along side it.


Promotion
Promotion_ID
Product_ID
Description
Promotion_Discount %
Start_Date
End_Date
Create_Promotion (Promotion_ID, Product_ID, Promotion_Discount %, Start_Date, End_Date)
Edit_Promotion ( )
Email_Campaign
Campaign_ID
Promotion_ID
Customer_ID
Campaign_Name
Create_Campaign (Campaign_ID, Promotion_ID, Campaign_Name)
Send ( )
Edit ( )
Submit ( )
Customer
Customer_ID
Customer_Fname
Customer_Lname
Email
Customer_Address
Customer_State
Customer_City
Customer_Zipcode
Customer_Telephone
Create_Customer (Customer_Fname, Customer_Lname, Customer_Email, Customer_Address, Customer_Telephone)
Submit ( )
Customer_Account
Customer_ID
Password
Newsletter
Security_Question
Create (Customer_ID, Password, Newsletter, Security_Question)
Edit_Account ( )
Forgot_Password ( )
Change_Password ( )
Review_Orders ( )
Appointment
Appointment_ID
Customer_ID
Quote request_ID
Appointment_Date
Appointment_Time
Create_Appointment (Appointment_ID, Quote_Request_ID, Appointment_Date, Appointment_Time)
Submit ( )
Delete_Event ( )
Event
Edit_Event ( )
Print_Schedule ( )
Premade_Dessert
Product_ID
Dessert_Name
Dessert_Price
Create_Dessert (Product_ID, Dessert_Name, Dessert_Price)
Delete_Dessert ( )
Edit_Dessert ( )
Custom_Dessert
Order_ID
Tiers
Servings
Cake_Flavor
Frosting_Flavor
Images
Notes
Frosting_Color
Filing_Flavor
Create_Custom Dessert (Order_ID, Servings, Cake_Flavor, Frosting_Flavor)
Upload_Image ( )
Order_Detail
Product_ID
Order_ID
Dessert_Price
Quantity
Discount
Create_Order Detail (Product_ID, Order_ID, Dessert_Price, Quantity)
Delete_Dessert ( )
Edit_Dessert ( )
Quote_Request
Quote Request_ID
Customer_ID
Order_ID
Appointment_ID
Custom_Desert *
Budget
Delivery
Customer_Address
Order_Status
Create_Quote Request (Quote Request_ID, Order_ID, Customer_ID, Customer_Address, Order_Status)
Select_Delivery ( )
Approve_Request ( )
Submit_Quote ( )
Reject_Request ( )
Login
Email
Password
Create_Login (Email, Password)
Forgot_Password ( )
Reset_Password ( )
Cash_Payment
Order_ID
Payment_ID
Customer_ID
Amount
Create_Cash Payment (Order_ID, Payment_ID, Amount))
Submit ( )
Transaction
Payment_ID
Transaction_ID
Transaction_Type
Create (Payment_ID, Transaction_ID, Transaction_Type)
Order
Order_ID
Customer_ID
Product_ID
Transaction_ID
Employee_ID
Item_ID
Quantity
Date Needed
Delivery
Address *
Paid
Create_Order (Order_ID, Customer_ID, Product_ID, Employee_ID, Item_ID, Quantity, Paid)
Select_Delivery ( )
Edit_Order ( )
Inventory_List
Item_ID
Item_Name
Item_Description
Unit_ofMeasurement
Unit_Price
Min_Quantity
Max_Quantity
Exp_Date
Product_ID
Create_Inventory List (Item_ID, Item_Name, Unit_Price, Min_Quantity, Max_Quantity, Product_ID)
Edit List ( )
Delete List ( )
Check_Payment
Payment_ID
Order_ID
Customer_ID
Amount
Create_Check Payment (Order_ID, Payment_ID, Customer_ID, Amount)
Submit ( )
Archive_Dessert
Order_ID
ImageOne_Thumbnail
ImageOne_JPEG
ImageTwo_Thumbnail
ImageTwo_JPEG
ImageThree_Thumbnail
ImageThree_JPEG
Tags
Create_Archive Dessert (Order_ID, ImageOne_JPEG, ImageOne_Thumbnail, Tags)
Delete_Image ( )
Add_Image ( )
Enlarge_Image ( )
Filter ( )
Search ( )
Marketing_Reports
Report_ID
Order_ID
Item_ID
Item_Name
Report_Date
Create_Marketing Report (Report_ID, Order_ID, Item_ID, Report_Date)
Print_Report ( )
Employee
Emp_ID
Emp_Fname
Emp_Middle Initial
Emp_Lname
Email
Emp_Social
Emp_Pay Rate
Emp_Hire Date
Emp_Title
Emp_Address
Emp_Zipcode
Emp_State
Emp_City
Create_Employee (Emp_ID, Emp_Fname, Emp_MiddleInitial, Emp_Lname, Emp_Social, Emp_Pay Rate, Emp_Hire Date, Emp_Address, Emp_Zipcdoe, Emp_State, Emp_City)
Edit ( )
Paypal_Payment
Member_ID
Transaction_ID
Order_ID

Entity Relationship Diagram (ERD)

How to read an ERD:

An ERD is a graphical representation of an information system that depicts the relationships among people, objects, places, concepts or events within that system. An ERD is a data modeling technique that can help define business processes and be used as the foundation for a relational database.



Archive Dessert

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Archive_Dessert	*A user who is viewing previously made products*	Order_ID	Unique id number for the order	String	4
		Image_Thumbnail	Unique id number for the customer	String	255
		Image_JPEG	Unique id number for the product	String	255
		Tags	Brief description of product	String	200

Custom Dessert

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Custom_Dessert		Order_ID	PK: Unique identification number for Custom_Dessert FK: connects to order table	String	4
		Tiers	The number of tiers for a cake, not required	Byte	1
		Servings	The number of servings needed by customer	Integer	4
		Cake_Flavor	One of the predefined available flavors offered	String	50
		Frosting_Flavor	One of the predefined available frosting flavors offered	string	50
		Frosting_Color	One of the predefined available Frosting flavors	String	50
		Filling_Flavor	One of the predefined available Filling flavors	String	50
		Images	List of image file paths	String	255
		Notes	Any applicable notes from the customer about their custom dessert	String	255

Customer

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Customer	*A user who is doing business*	Customer_ID	Unique id number for the customer	String	4
		Customer_Fname	First name of the customer	String	25
		Customer_Lname	Unique id number for the product	String	25
		Email	Unique id number for the payment	String	75
		Customer_Address	Unique id number for the employee	String	150
		Customer_State	Unique id number for the inventory item	String	2
		Customer_City	Number of products ordered	String	50
		Customer_Zipcode	Date for completion of the order	String	10
		Customer_Telephone	If delivery is requested	String	15

Customer Account

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Customer_Account	*A user who is creating an account*	Customer_ID	Unique id number for the customer	String	4
		Password	Password for logging into account	String	25
		Newsletter	If customer wants to receive newsletter	boolean	1
		Security_Question	Question in case password is forgotten	String	100

Customer Appointment

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Customer_Appointment	*An appointment set by the customer that the order should be completed by*	Appointment_ID	PK: Unique identification number for a customer's appointment	String	4
		Customer_ID	FK: Customer	String	4
		Quote_Request_ID	FK: Quote Request	String	4
		Appointment_Date	Date of appointment	Date/Time	8
		Appointment_Time	Time of appointment	Date/Time	8

Email Campaign

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Email_Campaign	*Used to hold templates (i.e welcome, birthday campaign, etc.)*	Campaign_ID	Unique identification number	String	4
		Promotion_ID	Foreign Key used to link the Promotion table	String	4
		Customer_ID	Foreign Key used to link the Customer table	String	4
		Camapign_Name	Name of the campaign	String	50

Employee

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Employee	*A user that will accept and fulfill orders*	Employee_ID	Unique identification number	String	4
		Employee_Fname	First name of employee	String	25
		Employee_MI	Employee middle initial	String	25
		Employee_Lname	Last name of employee	String	25
		Employee_Social	Employee social security number	String	9
		Email	Employee email address, doubles as user name	String	25
		Emp_Pay Rate	Current wage at which employee is paid	Decimal	4
		Emp_Hire Date	Employees first day of work	DateTime	8
		Emp_Title	Current title of employee (I.E. Manager, Supervisor, Temp Employee, etc.)	String	50
		Emp_Address	Current residence of employee	String	100
		Emp_Zipcode	Zip code employee lives in	String	5
		Emp_State	Employee State of residence	String	2
		Emp_City	Employee City of residence	String	50

Inventory List

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Inventory_List	*List of Inventory items that are used to create Karoline's products.*	Item_ID	PK: Unique identification number for premade dessert	String	4
		Item_Name	Name of the inventory item	String	25
		Item_Description	Description of inventory item	String	50
		Unit_ofMeasurement	Unit used to measure inventory item	String	25
		Unit_Price	Price of inventory item	Decimal	16
		Quantity	Amount of inventory item in stock	Integer	4
		Exp_Date	Expiration date of inventory item	String	6
		Product_ID	Optional identification number used only premade desserts	String	4

Login

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Login	*Holds the credentials of a user's account.*	Email	PK: users unique email address	String	255
		Password	The user's account password	String	255

Marketing Reports

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Marketing_Reports	*Pulls information from the order and item table to generate reports*	Report_ID	Unique identification number	String	4
		Order_ID	Foreign Key used to link the order table	String	4
		Item_ID	Foreign Key used to link the Item table	String	4
		Item_Name	Name of item (i.e. ingredient, utensil, cookware, etc.) being used	String	50
		Report_Date	Date the report was ran	DateTme	8

Order

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Order	*A user who is placing an order*	Order_ID	Unique id number for the order	String	4
		Customer_ID	Unique id number for the customer	String	4
		Product_ID	Unique id number for the product	String	4
		Transaction_ID	Unique id number for the payment	String	4
		Employee_ID	Unique id number for the employee	String	4
		Item_ID	Unique id number for the inventory item	String	4
		Quantity	Number of products ordered	Integer	2
		Date Needed	Date for completion of the order	Date/Time	8
		Delivery	If delivery is requested	Bit	1/8
		AddressLine1	Main line of the street address	String	100
		AddressLine1	Secondary line of the street address	String	50
		City	City of the address	String	50
		State	State of the address	String	2
		ZipCode	Zip code of the address	String	5
		Paid	If payment has been submitted	Bit	1/8

Premade Dessert

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Premade_Dessert	*Premade desserts that can be sold and delivered/picked up immediately*	Product_ID	PK: Unique identification number for premade dessert	String	4
		Dessert_Name	Name of premade dessert	String	25
		Dessert_Price	Price of each individual premade dessert	Decimal	16

Promotion

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Promotion	*A promotion sent out to customers offering discounts on products.*	Promotion_ID	PK: Unique identification number for promotions	String	4
		Product_ID	FK: Indicates the product being promoted	String	4
		Description	Description of what the promotion includes	String	255
		Promotion_Discount_%	Amount, in %, the promotion will discount the price	Decimal	5
		Start_Date	Date the promotion starts	String	6
		End_Date	Date the promotion ends	String	6

Quote Request

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Quote_Request	*An order for a custom dessert that starts as pending, and is either approved or rejected by the business.*	Quote_Request_ID	PK: Unique identification number for quote request	String	4
		Customer_ID	FK: Customer	String	4
		Order_ID	FK: Order	String	4
		Appointment_ID	FK: Appointment	String	4
		Custom_Dessert	FK: Custom_Dessert	String	4
		Customer_Budget	The specified budget specified by the customer	Decimal	16
		Delivery	Specifies whether the order is “delivery” (d), “undecided”(u), “not delivery” (n)	Bit	1/8
		Delivery_Address	Holds customer address as an Address Object	Special	255
		Order_Status	Holds the current state of the order: “pending”, “completed”, “cancelled”	String	9

Transaction

Entity Name	Entity Description	Column Name	Column Description	Data Type	Bytes
Transaction	*An exchange of currency for services rendered*	Transaction_ID	PK: Unique identification number of the transaction	String	4
			First name of employee		
		Payment_ID	FK: Indicates the type of payment	String	4
		Transaction_Type	Describes the type of transaction (Check, Cash, PayPal)	String	6

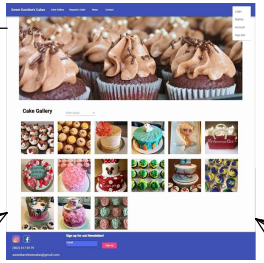
Website Navigation & Layout

The navigation of the Website is represented in a tree-like chart. At the top of the tree we find the first point of entry of the website, called the landing page. From the landing page we can go to many other parts of the website. This landing page shows where users can navigate to with a line connecting the two screens. From that screen we can see all of its possible navigation jumps we can make. In cases with account login being required the login screen comes before the page the user can access.

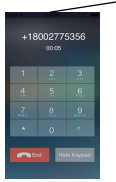
The first navigation chart is of the main Ecommerce site where most of the high-risk use cases live. The second navigation chart represents the back office and the pages of each of the tools required for the system. Such as Intuit quick books and the many pages that allow invoicing and business account settings.

Sweet Karoline's Cakes E-Commerce Website

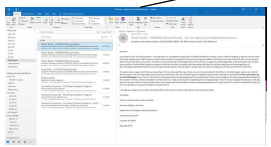
Admin Login



Landing Page and Cake Gallery

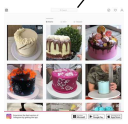


Cell Phone (Mobile)



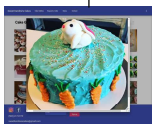
Email Application

Facebook

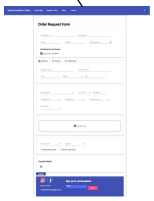


Instagram

Menu and Pricing



Enlarged Photo Modal



Order Request Form

PayPal

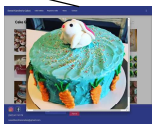
Calendar Modal

Contact Us

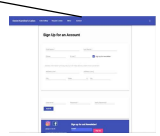
Login

Account Details

Edit Order Request



Enlarged Photo Modal



Sign Up for an Account

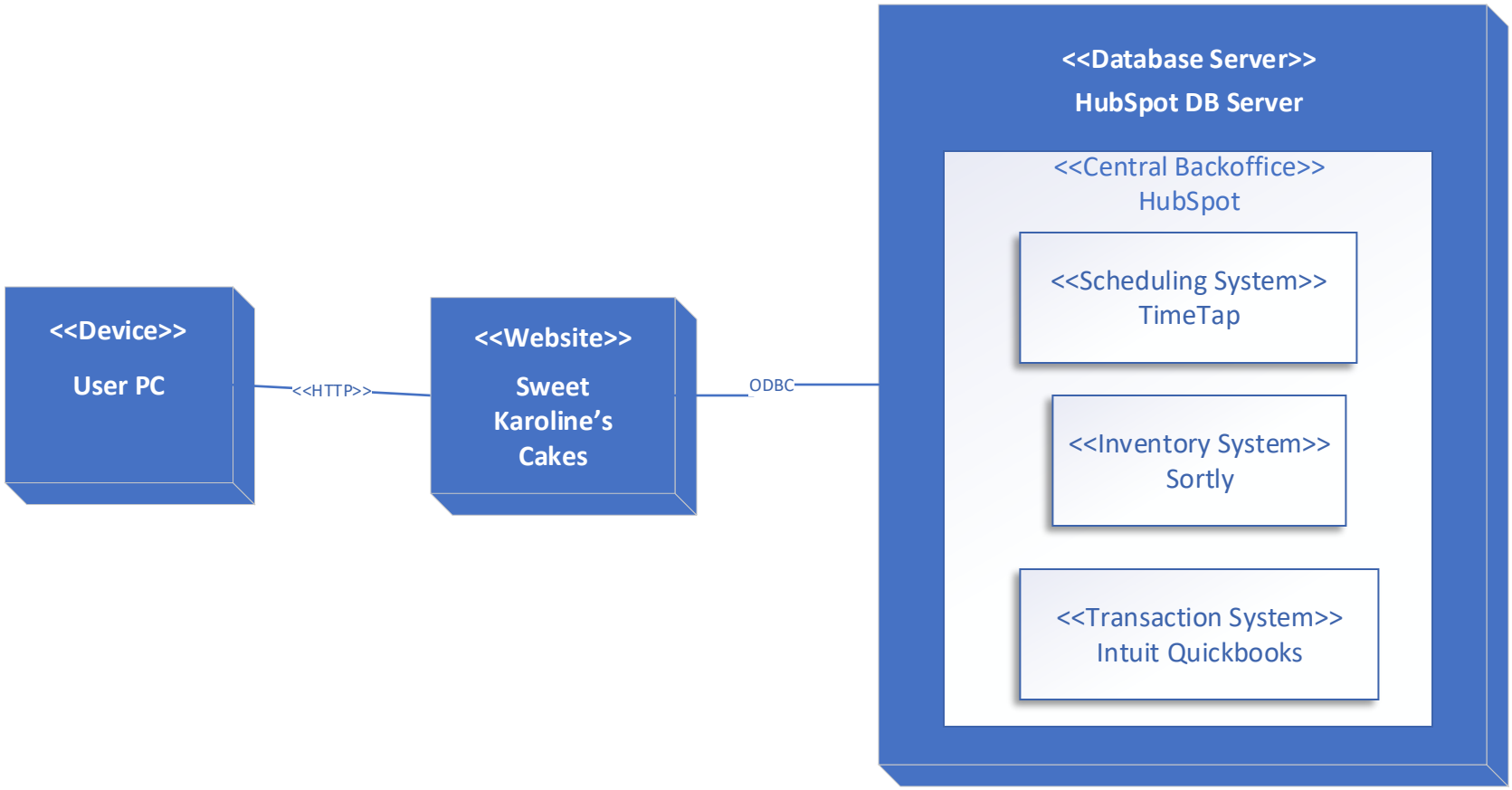
ADMIN Back Office

Admin Login

- WordPress CMS
  - HubSpot
    - QuickBooks
      - Dashboard
      - Account Settings
        - Company
        - Billing & Subscription
        - Sales
          - Customize Invoice
        - Expenses
        - Payments
        - Advanced
      - Sales
      - Expenses
      - Employees
      - Reports
      - Taxes
      - Banking
        - Connect Account
      - Apps
      - My Accountant
      - Accounting
    - Hubspot CRM
      - Add Contact
      - Add Filter
      - Contact Details
        - Schedule
        - New Note
        - Email
        - Log Activity
        - Call
        - Create Task
      - Banking
      - Banking
    - Sortly
      - Settings
        - Manage Users
      - Item Folder
        - Export
        - Edit Item
        - Move Item(s) Modal
  - TimeTap Scheduling
    - Create Appointments
    - Manage Appointments
      - See Appointment Details
        - Email Contact
      - Edit Appointment
    - Print Schedule

Physical Architecture & Design

The user accesses the website through whatever device fits their needs at that time, but for general purposes we are assuming they are using a personal computer or laptop. They connect to the CMS through an HTTP portal on the internet and get onto the Sweet Karoline's Cakes website. The website connects to the HubSpot database server through an open database connectivity portal, and the database holds the software applications used for the functionalities of the website (TimeTap, Sortly, and Intuit Quickbooks).


<<Device>>
User PC
<<HTTP>>
<<Website>>
Sweet
Karoline's
Cakes
ODBC
<<Database Server>>
HubSpot DB Server
<<Central Backoffice>>
HubSpot
<<Scheduling System>>
TimeTap
<<Inventory System>>
Sortly
<<Transaction System>>
Intuit Quickbooks

Design Procedures for Security Concerns

The following section will describe the procedures associated with the concerns of security.

Customer login
Each customer must be able to access only their individual information. The users account will be secured with a customized username and password. The customer must login with their credentials before making any changes to their account info or orders.

Back-office access
The back-office software will require the administrator to provide a user login and password to be able to access the systems. This will keep all unauthorized individuals from being able to change or acquire the information.

Scheduling
The schedule will require an administrator login to be able to add, change, or cancel the completion dates. The customers will initiate requests for changes which will require approval. The schedule availability dates will also require administrator login to make changes.

Payments
All payments will be processed in person or online through PayPal. PayPal is responsible for ensuring the security and validity of all online payment. In the event of an error or discrepancy PayPal will act as intermediary to ensure the issue gets resolved.

Data Backup
The data will be backed-up automatically each day at the close of business. The database software will perform this automatically and provide a notification when it is completed. The administrator will be able to manually update at any time but must be logged into the system to perform the action. The Database system will be responsible for providing encryption as well as security for the backed-up data.

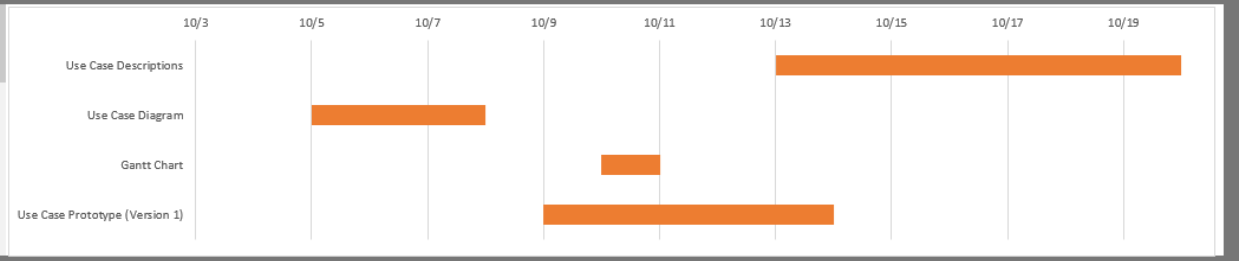
Data Recovery
The data will be able to be recovered at any time from the backup stored by the database system. A bi-annual test of the recovery process will be administrated by the administrator. The recovery test will be conducted by logging into the database system with the proper credentials and attempting to download the backed-up data. The data obtained by the recovery process must then be checked to endure the information is correct and up-to-date.

Gantt Chart

Below are the Gantt charts for Iteration 3 and Iteration 5. They depict the deliverables we have completed during the iterations, along with a graphical timeline. These provide the team with timelines for who should be completing which deliverable and when they should be completed. This allowed us to hold each other accountable and ensure that each part of the iteration was successfully completed.

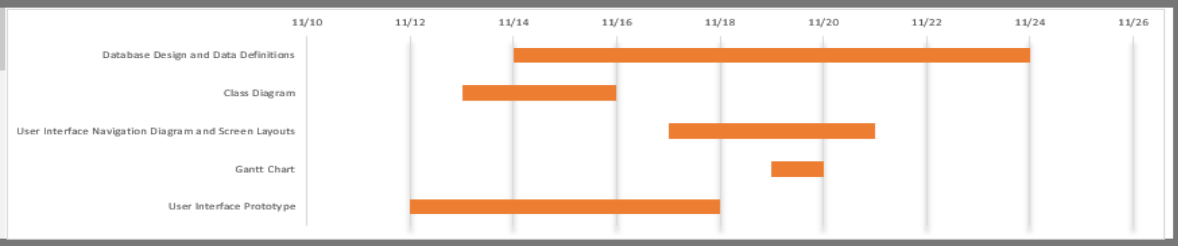
Iteration 3

START DATE	END DATE	DESCRIPTION	Assigned To	DURATION (days)
10/13/18	10/20/18	Use Case Descriptions	Kevin Bush, Logan Amlung, Brad Eadesn, Hannah Parker, Ben Lascurain, Chris Boaz	7
10/5/18	10/8/18	Use Case Diagram	Logan Amlung/ Brad Eadens	3
10/10/18	10/11/18	Gantt Chart	Hannah Parker	1
10/9/18	10/14/18	Use Case Prototype (Version 1)	Ben Lascurain/Chirs Boaz	5



Iteration 5

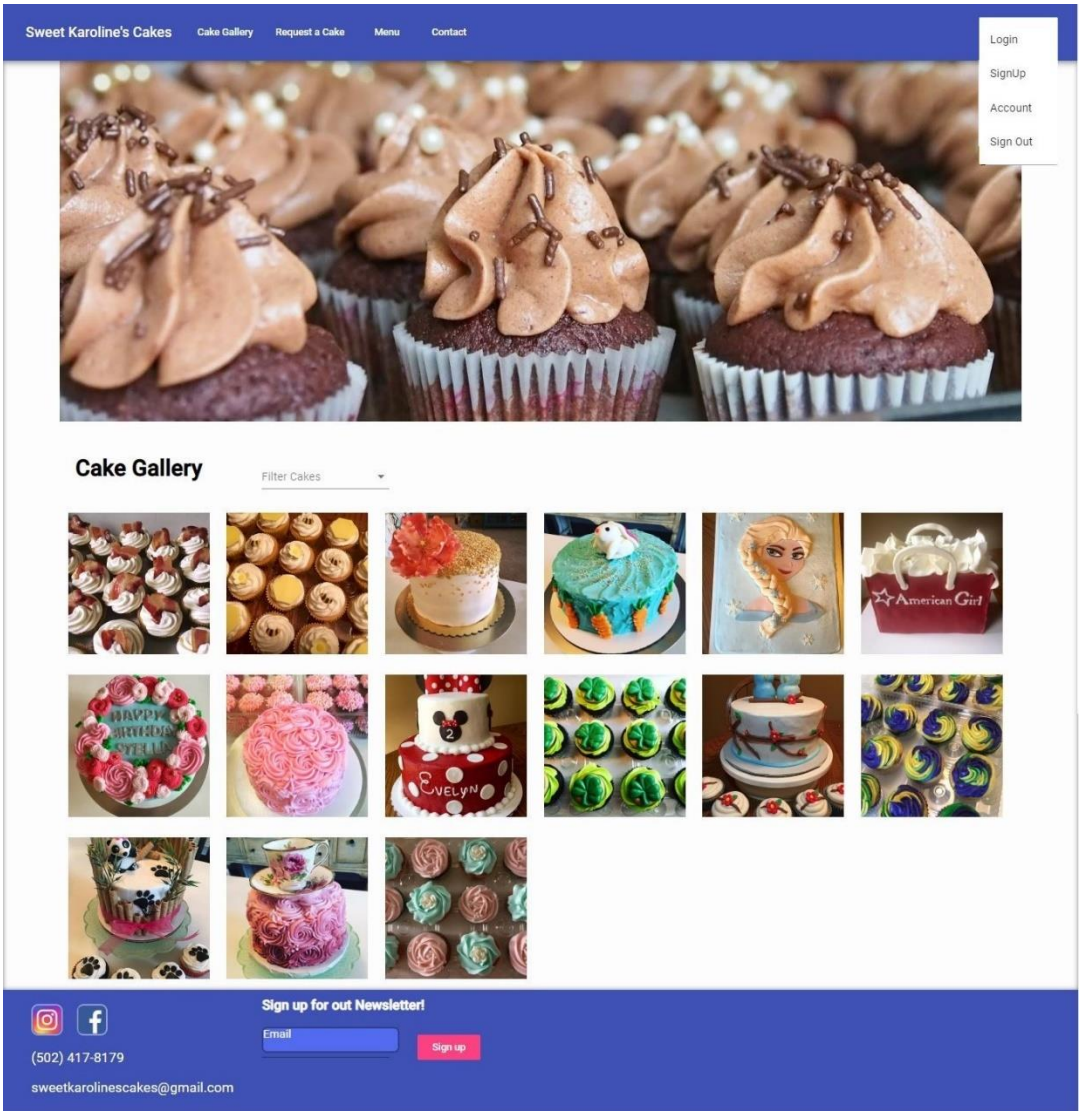
START DATE	END DATE	DESCRIPTION	Assigned To	DURATION (days)
11/14/18	11/24/18	Database Design and Data Definitions	Kevin Bush, Logan Amlung, Brad Eadesn, Hannah Parker, Ben Lascurain, Chris Boaz	10
11/13/18	11/16/18	Class Diagram	Brad Eadens	3
11/17/18	11/21/18	User Interface Navigation Diagram and Screen Layouts	Ben Lascurain	4
11/19/18	11/20/18	Gantt Chart	Hannah Parker	1
11/12/18	11/18/18	User Interface Prototype	Ben Lascurain	6



Website Prototype

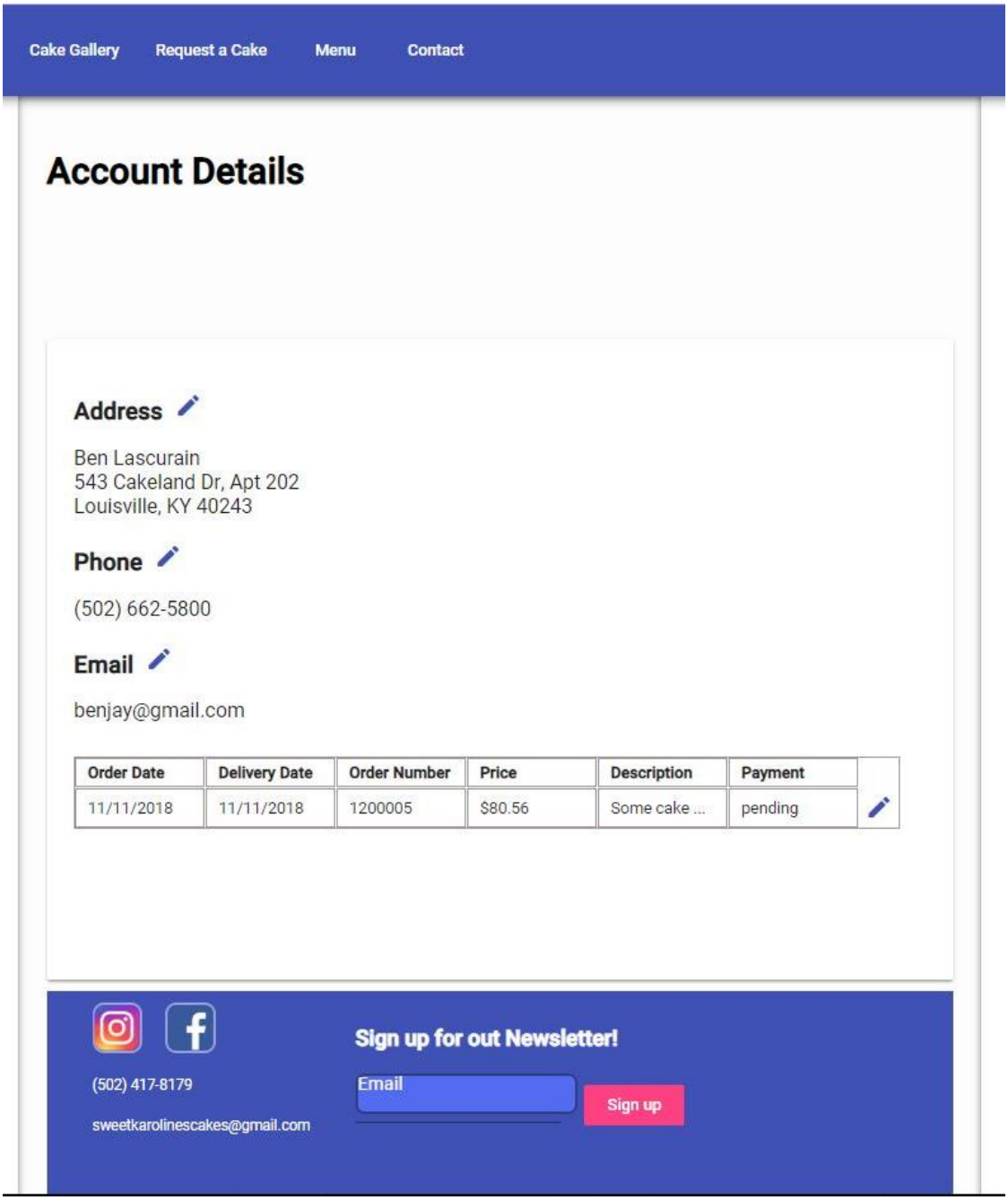
Cake Gallery Narrative:

Users want to be able to see examples of previous custom cakes and filter through products based on categories. Featured below is a basic gallery with a filter feature in the upper right.



Customer Account Management Narrative:

Users need to be able to manage their account information and see their past orders. This should be accessible after logging in with a username and password. Having an account is important for repeat customers, so that their checkout process will be much faster and will encourage repeat orders.

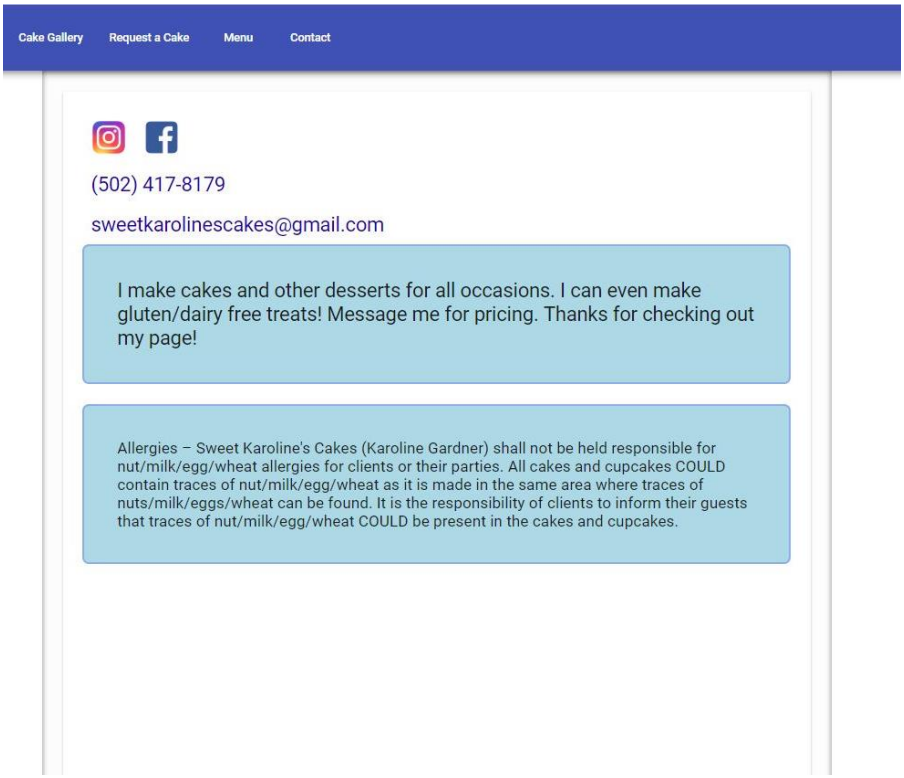


Edit Order Request:

Users should be able to request modifications to their custom orders. Customers should be able to add a delivery address and date. Customers should be able to make payments with PayPal.

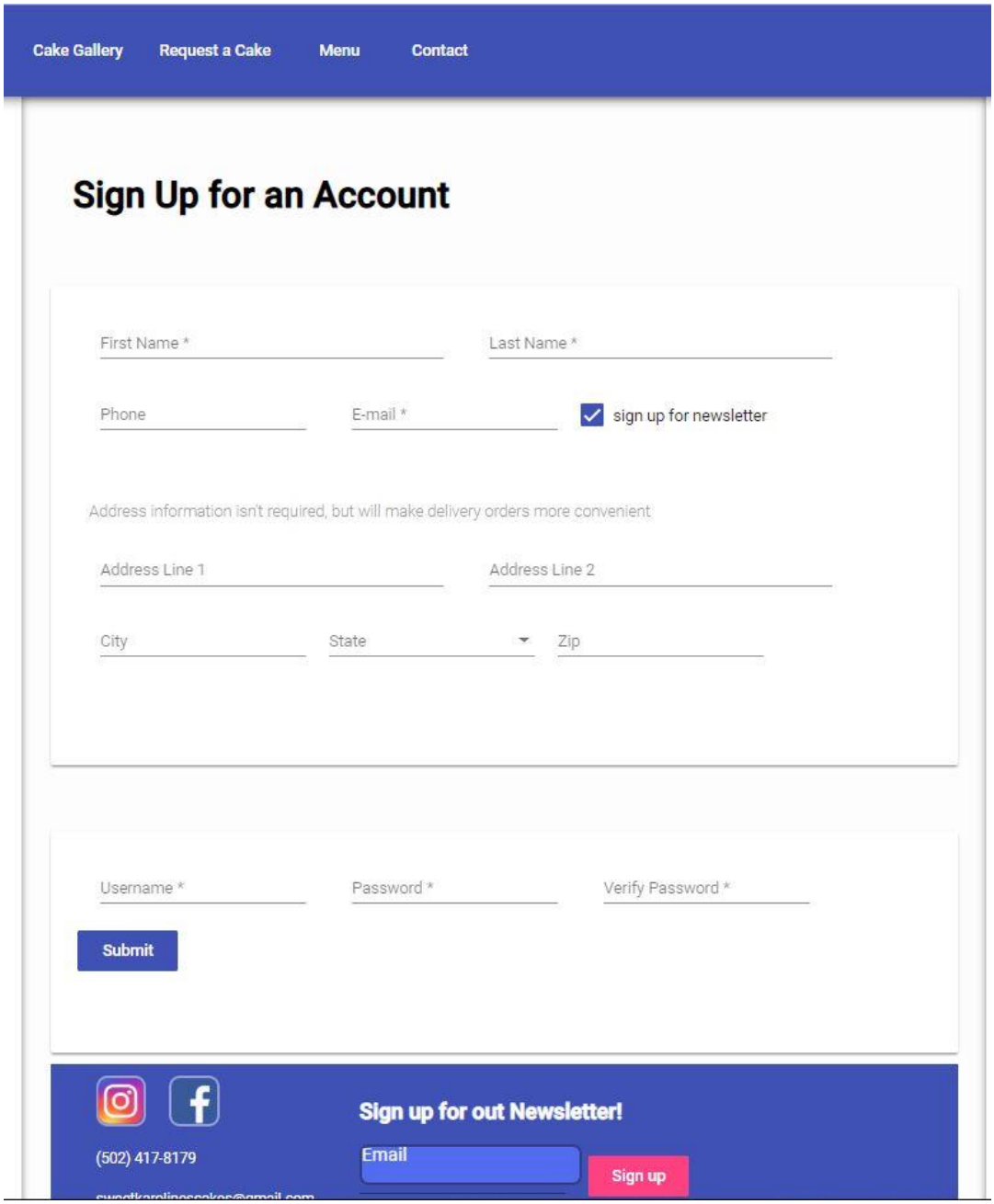
Contact Us Page:

Customers want to be able to find contact information and Social Media information.



Customer Account Sign-Up Narrative:

Users should be able to create an account with pertinent ordering information. This information should cover the basis of delivery orders, contact information, payment methods (which should be limited to PayPal for lability), and username and password information to make the account secure and retrievable.

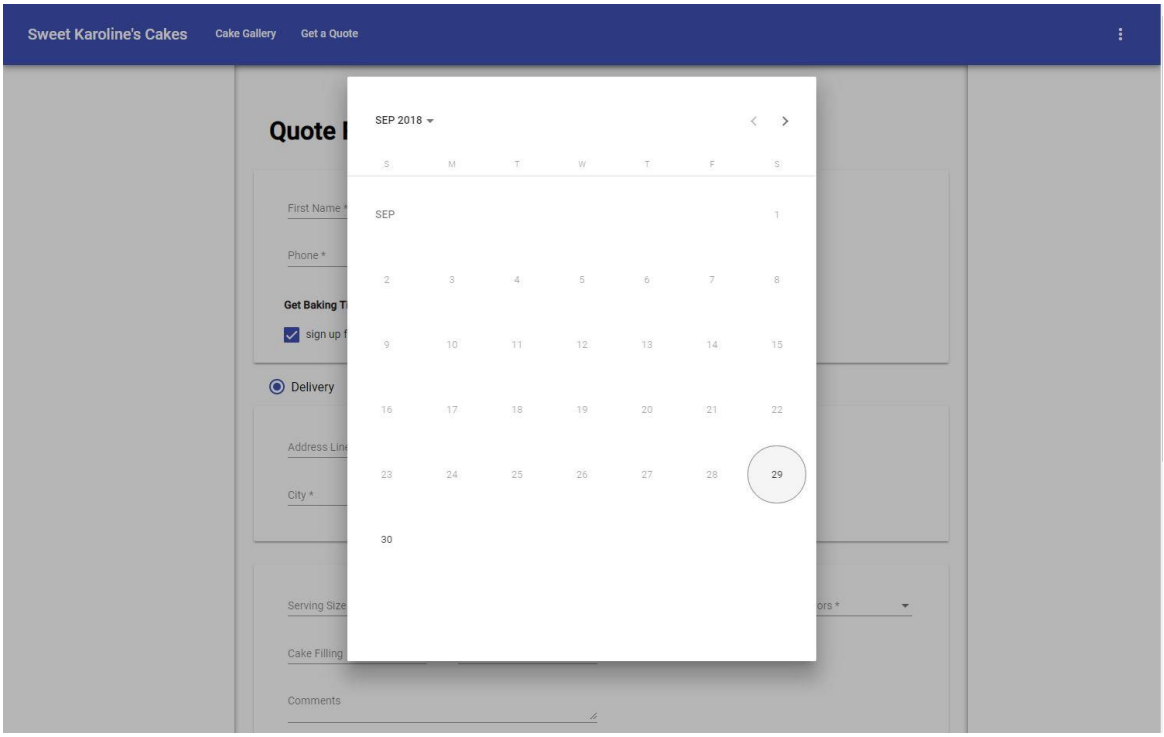


Customer Login Narrative:

Users should be able to login to their accounts and have their information and orders stored privately. Having a login accomplishes this.

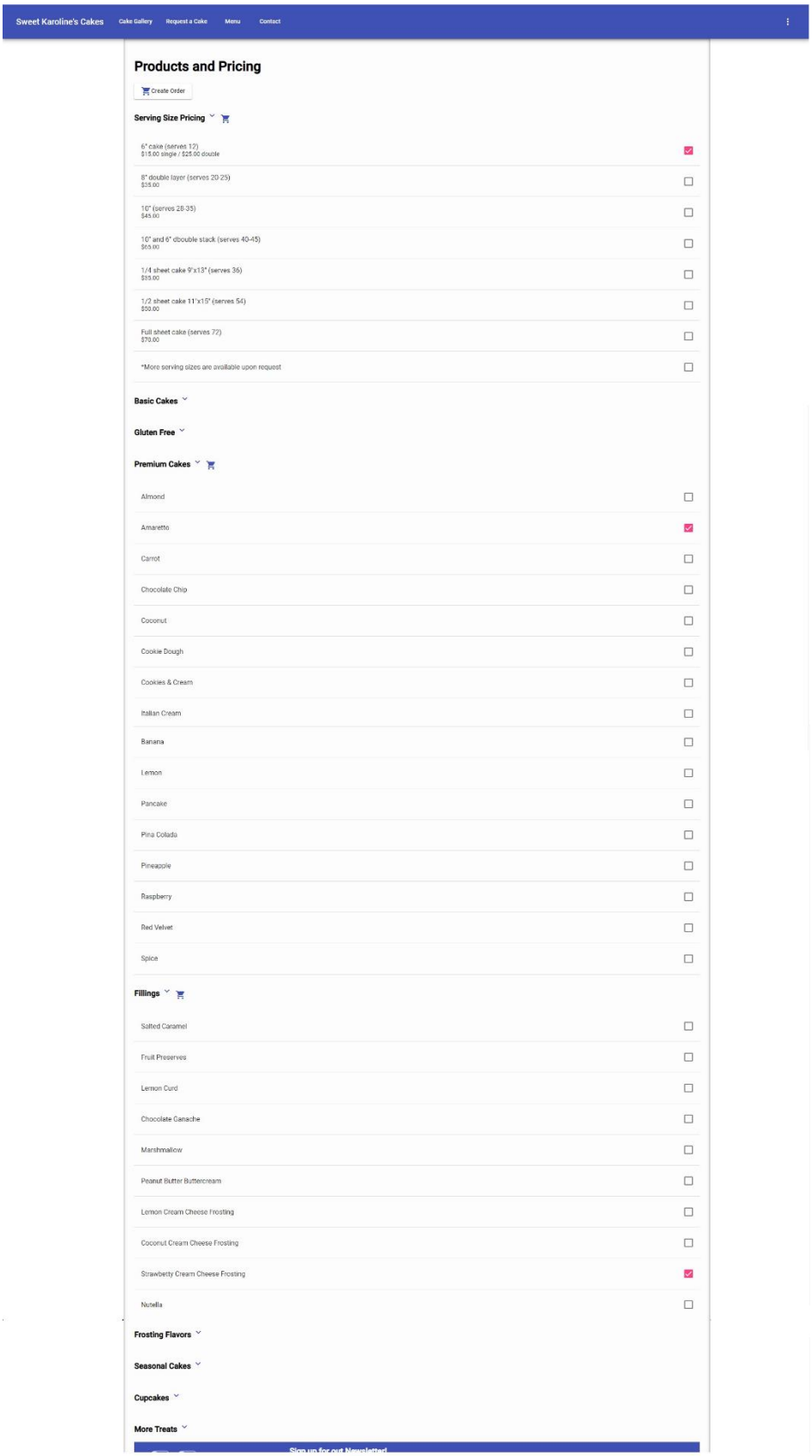
Caledar Availiability Narrative:

Customers should have order availiability available to them in a self sufficient manner. The days in which Karoline is unavailable should be greyed out and disabled as invalid date options.



Menu and Pricing:

Users want to be able to see all available products to purchase and their correlating prices. The products and pricing page will fill this need.



Ordering Narrative:

Customers will need to able to create order requests. In order to do this they need a standardized and complete way to request products while conveying specifics that would need to be known to by Karoline to complete the order. This would require Name, Contact information, requested order date, delivery/pickup, address for delivery cake details such as serving size, tiers, cake flavor, frostings, an image drop for cake designs, and other pertinent cake details. This order request from should also have payment information. The customer information should be autofilled if the customer has an account.